V m

2

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

AD-A243 065

...o average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and ...n. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions ...torate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to ...t and Budget, Washington, DC 20503.

2. REPORT DATE Dec 91	3. REPORT TYPE AND DATES COVERED Thesis/Final

4. TITLE AND SUBTITLE
Characterization of the Thermalness of a Fissile System with a 2-Group Diffusion Theory Parameter

5. FUNDING NUMBERS

6. AUTHOR(S)
Brent B. Bredehoft, CPT USA

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
University of New Mexico
Department of Chemical and Nuclear Engineering
Albuquerque, NM 87131

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Total Army Personnel Command
ATTN: TAPC-OPB-E
200 Stoval Street
Alexandria, VA 22332-0411

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES
Prepared as a requirement under the Fully Funded Advanced Civil Schooling Program IAW AR 621-7.

12a. DISTRIBUTION/AVAILABILITY STATEMENT
Unlimited

12b. DISTRIBUTION CODE

13. ABSTRACT (Maximum 200 words)

Tabulation of critical data has commonly been done using the hydrogen to fissile atom ratio [H/X]. This does not account for the moderating contribution of other light nuclei in the system or for the absorption characteristics, important to the slowing down process.

Using a two energy group diffusion theory analogy to the six-factor-formula, we define the parameter $p/(\eta_2 {}^* f_2)$ as an appropriate parameter for defining moderation characteristics. This parameter adequately accounts for the important details of the slowing down process; resonance absorption, total system scattering and moderation, and the effective use of thermal neutrons.

This study evaluates several low to moderate enriched uranium systems with several types of moderators. It utilizes the TWODANT transport code with the Hansen-Roach 16-group cross sections and group collapse theory to develop appropriate system models and 2-group cross sections.

14. SUBJECT TERMS
Thermal Uranium System, Diffusion Theory, Moderation, Thermal, Group Collapsing of Cross Sections, 2-Group Theory

15. NUMBER OF PAGES
158

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	Unlimited

NSN 7540-01-280-5500

Standard Form 298, 880922 Draft
Prescribed by ANSI Std. 239-18

Accession For

NTIS GRA&I ☑

DTIC TAB ☐

Unannounced ☐

Justification

By

Distribution/

Availability Codes

Dist	Avail and/or Special
A-1	

Brent Bryce Bredehoft

Candidate

Chemical and Nuclear Engineering

Department

This thesis is approved, and it is acceptable in quality and form for publication on microfilm:

Approved by the Thesis Committee:

Robert D Busch, Chairperson

R Dougl O'Dell

Accepted:

Asst. *Dean, Graduate School*

November 19, 1991

Date

91-16873

91 12 02 039

Characterization of the Thermalness of a Fissile System with a 2-Group Diffusion Theory Parameter

BY

BRENT BRYCE BREDEHOFT

B.S., United States Military Academy, 1983

THESIS

Submitted in Partial Fulfillment of the
Requirements for the Degree of
Master of Science in Nuclear Engineering

The University of New Mexico
Albuquerque, New Mexico
December 1991

ACKNOWLEDGEMENTS

I wish to thank Robert Busch for the intitial thesis topic and the continued support and guidance throughout this study. None of this was possible with out his dedication. Additionally, R. Douglas O'Dell from Los Alamos National Laboratory provided immeasurable assistance and guidance in all areas of this project. Of course, the entire thesis committee deserves praise for their valuable feedback. Finally, my wife, Shari and my children deserve a great deal of credit for tolerating the long hours and mood swings associated with this entire process.

Characterization of the Thermalness of a Fissile System with a 2-Group Diffusion Theory Parameter

BY

BRENT BRYCE BREDEHOFT

ABSTRACT OF THESIS

Submitted in Partial Fulfillment of the
Requirements for the Degree of
Master of Science in Nuclear Engineering

The University of New Mexico
Albuquerque, New Mexico
December 1991

Characterization of the Thermalness of a Fissile System with a 2-Group Diffusion Theory Parameter

Brent Bryce Bredehoft

B.S., United States Military Academy, 1983

M.S. Nuclear Engineering, Univ. of New Mexico, 1991

In tabulating critical data, the hydrogen to fissile atoms ratio, H/X, is commonly used to specify the amount of moderation in a system. Though adequate in many cases, H/X does not account for the moderating contribution of other light nuclei contained in common uranium-moderator mixtures. This ratio also does not account for enrichment of the system, which affects the resonance absorption characteristics and, therefore, the moderating characteristics of that system.

We used a two energy group diffusion theory analogy to the six-factor formula to define the parameter $p/(\eta_2 * f_2)$ as appropriate for describing the moderation characteristics, or the "thermalness" of a fissioning system. The neutron slowing down process and system enrichment are adequately described by the resonance escape probability, p. The absorption characteristics of the system, particularly the effectiveness of neutrons in causing fission versus non-fission capture, are encompassed in the η_2 and f_2 factors. Therefore, the important details in describing

moderation are contained in the thermalness factor $p/(\eta_2{}^*f_2)$. Plots of the thermalness factor versus critical mass and volume serve to predict minimum critical mass and volume for moderated systems and indicate optimum moderation characteristics.

In this study we evaluated several low enriched uranium systems with different hydrogenous moderators. The systems were originally modeled using the transport theory code, TWODANT with the Hansen-Roach 16-group cross section set. The 16-group cross sections were group collapsed, using TWODANT derived fluxes, into the 2-group cross sections used for determining $p/(\eta_2{}^*f_2)$.

From our analysis we found that the values of $p/(\eta_2{}^*f_2)$ have a narrower range than the values of H/X corresponding to minimum critical mass and volume for different systems. Also, $p/(\eta_2{}^*f_2)$ does not vary with the addition of a reflector and is applicable to systems with other than hydrogenous moderators. Based on these results, the thermalness parameter $p/(\eta_2{}^*f_2)$ provides an effective means of characterizing moderated systems in terms of optimum conditions.

TABLE OF CONTENTS

APPENDICES

LIST OF FIGURES

LIST OF TABLES

1. INTRODUCTION.

In any system containing fissile material, a very important consideration is the amount of moderation. Moderation is the act of slowing down neutrons and is a multifaceted process, depending on the type and amount of scattering material in the system. However, the effectiveness of a moderator also depends on a lack of absorption during the scattering process. This suggests that the "thermalness" of a system depends on the amount and type of fuel as well as the moderator characteristics.

In tabulating critical data, the hydrogen to fissile atom ratio [H/X] is commonly used to specify the amount of moderation in a system. Though adequate for systems with hydrogen, H/X does not take into account the moderating contribution of other light nuclei contained in the system. Unquestionable is hydrogen's superior moderating capability, but carbon, oxygen, and several other light nuclei contained in common uranium-moderator mixtures do contribute to the slowing down of neutrons. For example, sterotex (glycerol tristearate or $(C_{17}H_{35}COO)_3C_3H_5$) and polyethylene ($CH_2$) have carbon in sufficient quantity to greatly add to neutron moderation. Other mixtures such as U_3O_8-sterotex have oxygen in enough quantity to provide additional moderation.

H/X also neglects the enrichment of the fuel. Enrichment specifies, for one thing, how much U^{238} is in the system. This is a very important parameter in moderated systems since neutrons have to escape the large resonance absorption of U^{238}. Similarly, H/X tells us nothing about any kind of absorption in the system. The effectiveness of a moderator is a measure of its ability to scatter without absorbing neutrons. Expand this to a complete, moderated system and the effectiveness is then a measure of scattering from all isotopes in the system without unfavorable absorption (a favorable absorption is a fission interaction, though a thermal fission is preferred to a fast fission).

This study looks at the use of parameters, other than H/X, to describe the "thermalness" of systems, where we describe "thermalness" as the degree of moderator effectiveness. It specifically tries to find a parameter which will be useful in predicting a minimum critical volume and mass for different fuel-moderator mixtures. As with tabulation and plotting of critical experiments based on H/X, examples of which are shown in Figures 1 and 2, our parameter must have the parabolic shape that leads to a minimum critical mass and volume. In Figures 1 and 2 we see a that the minimum critical volumes and masses appear over a wide range of H/Xs. Apparently, minimum volume and mass are a widely varying function of H/X, enrichment, and type of fuel-moderator mixture. The objective of this effort is to identify a parameter which will narrow this range and remove the dependence on enrichment and type of fuel-moderator mixture.

Predicting minimum critical volumes and masses has become a relatively simple task with the use of modern computers and computer codes. However, we desire a simple parameter that can be readily and accurately derived by hand calculations. It is hoped that such a parameter will provide insight into the character of a very thermal system with a highly effective moderator.

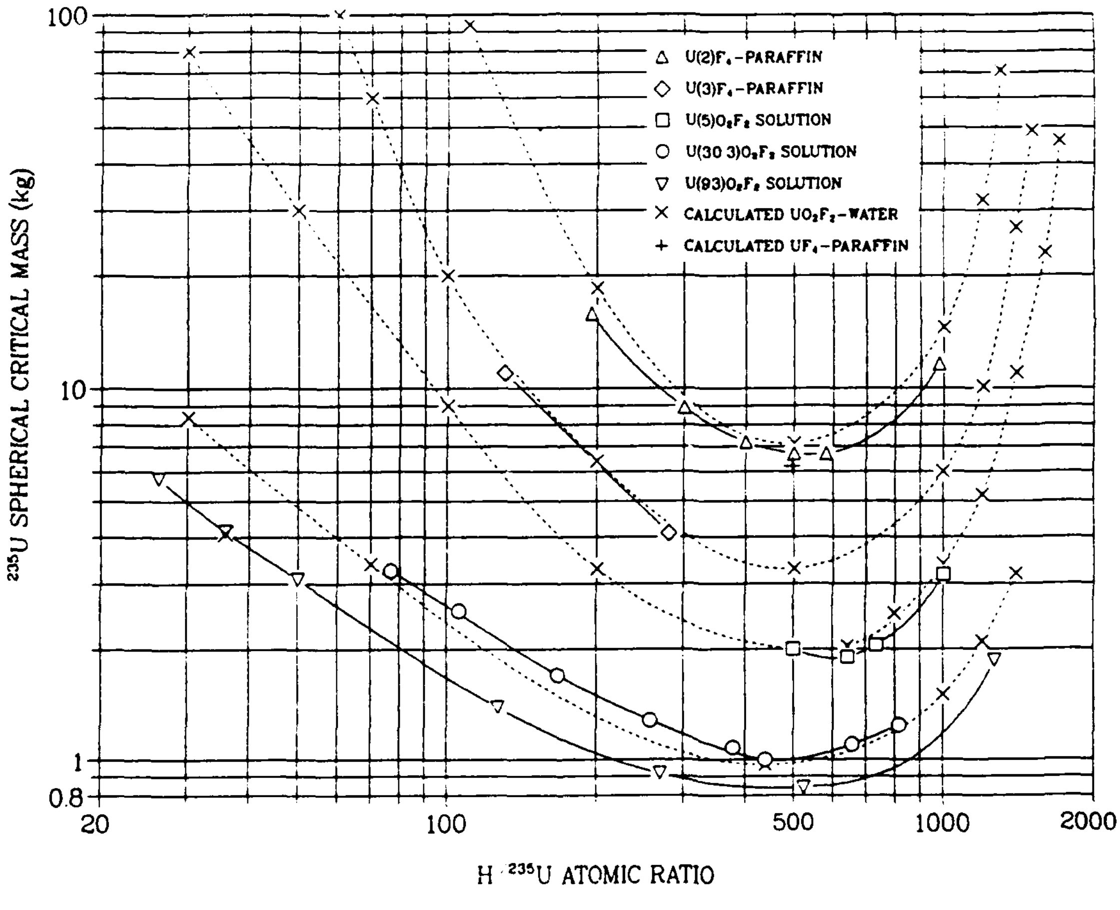

Figure 1, Critical masses of water-reflected spheres of hydrogen-moderated U(93), U(30.3), U(5.00), U(3.00), and U(2.00). [From p.37, Reference 1].

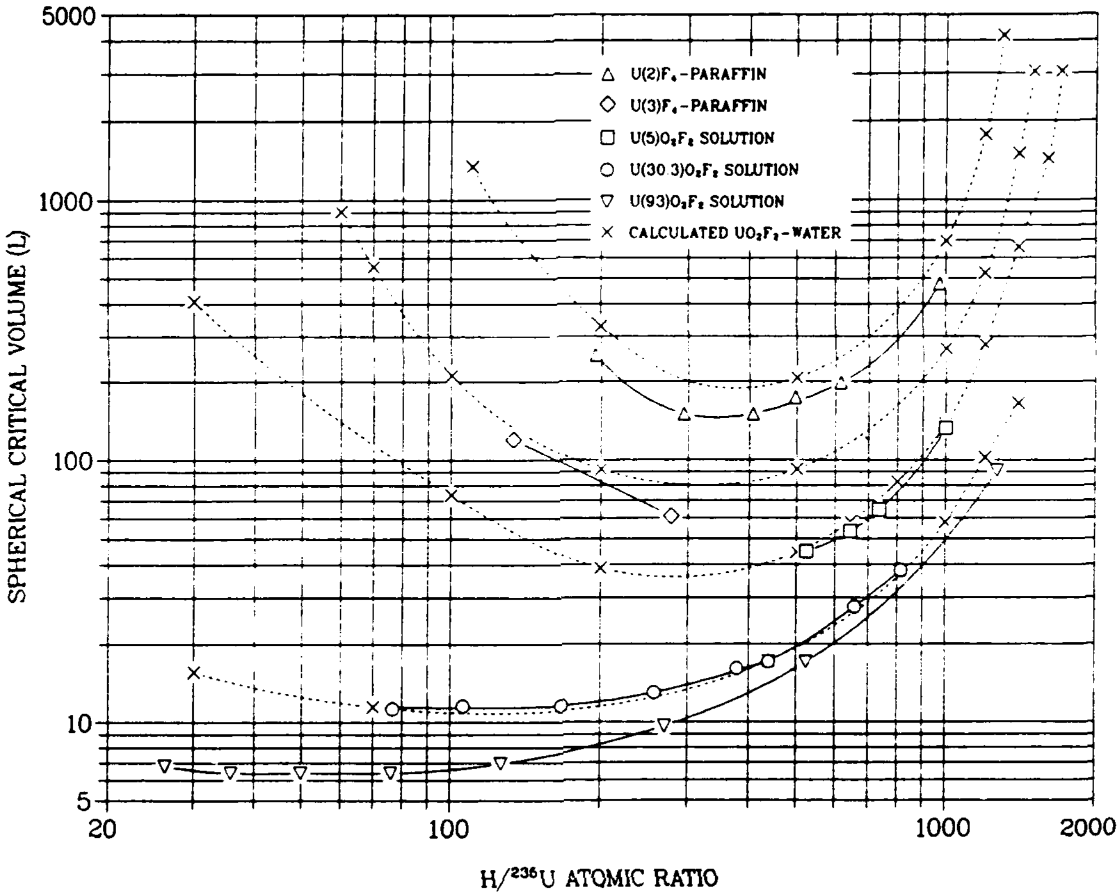

Figure 2, Critical volumes of water-reflected spheres of hydrogen-moderated U(93), U(30.3), U(5.00), U(3.00), and U(2.00). [From p.38, Reference 1].

As will be explained in detailed in this report, we have selected a 2-group diffusion theory analogy to the six-factor formula for calculating the parameter $p/(\eta_2 {}^* f_2)$, which meets all of the criteria desired in describing moderating capability. The 2-group cross sections are group collapsed from the Hansen-Roach 16-group cross section set. The transport theory code TWODANT is used to develop the weighting functions and spectrum calculations needed for the group collapses.

2. DEVELOPMENT OF THE PARAMETER TO DESCRIBE MODERATING CAPABILITY.

Moderation is the process of slowing down neutrons from fission energies (100KeV-10MeV) to thermal energies (on the order of 10^{-2} eV). The scattering interaction, where neutrons loose a fraction of their energy, is the primary mechanism for this process. However, during the slowing down process there are ways that the neutrons can be lost and this is also an important effect in describing how well a system is moderated. The main method of neutron loss during slowing down is absorption. No matter how well a system does in scattering and hence slowing down neutrons, if the neutron is absorbed in a non-fissioning interaction the system is not well moderated. Resonance absorption is one of the biggest obstacles in neutron moderation. The large magnitude of the resonance absorption cross sections makes it very important to consider how well this is avoided when describing moderated systems. In uranium systems U^{238} is the dominate resonance absorber. Therefore, when discussing the moderation process the amount of U^{238}, a factor of enrichment, is a very important consideration. With these factors in mind we will now select a parameter that encompasses all of the important characteristics in describing the moderating capability of uranium systems.

In describing moderation, other parameters besides H/X are often used to compare moderators. Two of these are:

$$\text{Moderating power} \equiv \xi\Sigma_s \tag{1}$$

and

$$\text{Moderating ratio} \equiv \frac{\xi\Sigma_s}{\Sigma_a} \tag{2}$$

These were developed primarily to compare moderators but can be used to compare complete systems if the lethargy gain, ξ, and the macroscopic cross sections, Σ, represent total system values rather than moderator values.

Moderating power does not incorporate the absorption characteristics of the moderator or system and is, therefore, incapable of meeting our requirements. The moderating ratio does consider absorption and in a sense also considers enrichment as the Σ_a will reflect the isotope concentrations. However, as a one energy group parameter, it does not adequately describe interactions in the resonance region.

Looking for an easily calculable parameter that will adequately address scattering, absorption, and ability to escape resonance absorption, we are drawn to the six-factor formula for K_{eff}. The six-factor formula is a one group analysis, but provides parameters which attempt to describe the interactions at fast energies and in the resonance region. The factors in this formula,

η = number of neutrons produced per neutron absorbed in fuel, (3)

f = utilization factor (number of neutrons absorbed in fuel per neutron absorbed in the system), (4)

ε = fast fission factor (number of neutrons produced by fissions from fast and thermal neutrons per number of neutrons produced by fissions from thermal neutrons), (5)

p = resonance escape probability (probability that a fission neutron successfully slows down to thermal energies), (6)

PNL_f = probability of non-leakage of fast neutrons, (7)

PNL_t = probability of non-leakage of thermal neutrons, (8)

cover all of the requirements we are looking for. The resonance escape probability is a measure of how important resonance absorption is in the system. η and f tell us how effectively we use the thermal neutrons and therefore account for the absorption and production properties of the system. The non-leakage probabilities indirectly express the effectiveness of the system at slowing down neutrons. The more moderation taking place the less the chance is the neutron will leak prior to being absorbed.

The six-factor formula, however, is based on a one group model. A one group model is not always appropriate for determining moderation characteristics of a system, since by definition, moderation involves the slowing down of neutrons from higher to lower energies. It is probably appropriate for true thermal systems and for comparing sizes of systems for various moderators. As a first approximation it is very good. For a better description of the process, a multigroup approach seems neccessary. However, using more than two energy groups makes the six-factor formula cumbersome and effectively useless. What we will do is use a two-group diffusion theory approach and draw an analogy to the six-factor formula. This results in definitions of the factors for the six factor formula based on two group variables.

2.1 Derivation of the Two Group Diffusion Theory Equations.

The steady state multigroup diffusion equation is,

$$-\nabla \cdot D_g \nabla \phi_g + \Sigma_{Rg}\phi_g = \sum_{g'=1}^{g-1} \Sigma_{sg'-g}\phi_{g'} + \frac{1}{k}\chi_g \sum_{g'=1}^{G} \nu_{g'}\Sigma_{fg'}\phi_{g'}, \tag{9}$$

where D_g = the diffusion coefficient for group g,

Σ_{Rg} = the Macroscopic removal cross section for group g ($\Sigma_{Rg} = \Sigma_{tg} - \Sigma_{sg-g}$),

$\Sigma_{sg'-g}$ =the macroscopic in-scatter cross section for group g from group g',

k = the multiplication factor,

χ_g = the fraction of fission neutrons created whose energies lie within group g,

$\nu_{g'}$ = the average number of fission neutrons from each fission event for group g',

Σ_{fg} = the macroscopic fission cross section for group g,

and we have assumed no upscattering occurs.

For the development of the two group analysis, we define

$$\phi_1(r) = \int_{E_1}^{E_0} dE\phi(r,E) \equiv \text{FAST GROUP FLUX}\,, \tag{10}$$

$$\phi_2(r) = \int_{E_2}^{E_1} dE\phi(r,E) \equiv \text{THERMAL GROUP FLUX}\,, \tag{11}$$

where E_0, E_1, and E_2 are the maximum neutron energy, cut off energy between groups, and minimum neutron energy respectively. For a fissioning system we define E_0 as 10 MeV and E_2 as 0. The development of the value for E_1, the separation energy between the fast and thermal groups, is based on two factors. The first is that E_1 needs to be high enough so up scatter out of the thermal group can be ignored. For most systems this corresponds to a value between 0.5 and 1.0 eV. The second is that E_1 should be low enough to ensure that the neutron cross sections in the thermal group are well behaved (i.e., a 1/v behavior and no resonance peaks). In this work we are primarily considering uranium systems moderated with hydrogen or carbon compounds and possibly containing other low Z elements such as fluorine, nitrogen, and oxygen. Noting the behavior of these materials' cross sections (Figures 3 through 8) we have selected the cut off energy between the two groups, E_1, as 1.0 eV. Note that the U^{235} and U^{238} cross sections are the specific ones driving the selection of the cutoff energy due to their resonance characteristics.

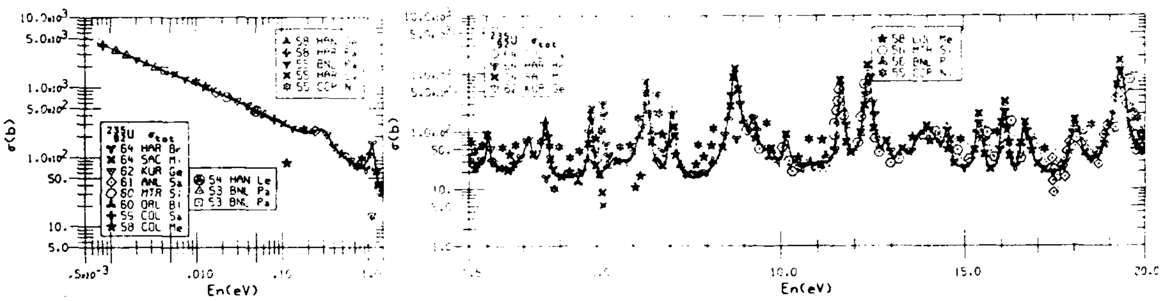

Figure 3, Total cross section of U^{235}. [From p. 446, Reference 2].

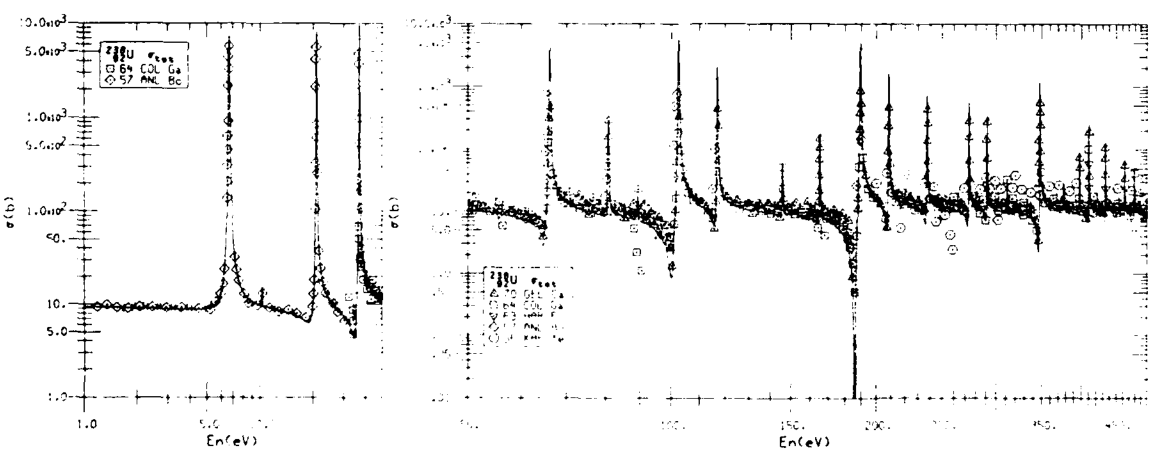

Figure 4, Total cross section of U^{238}. [From p. 454, Reference 2].

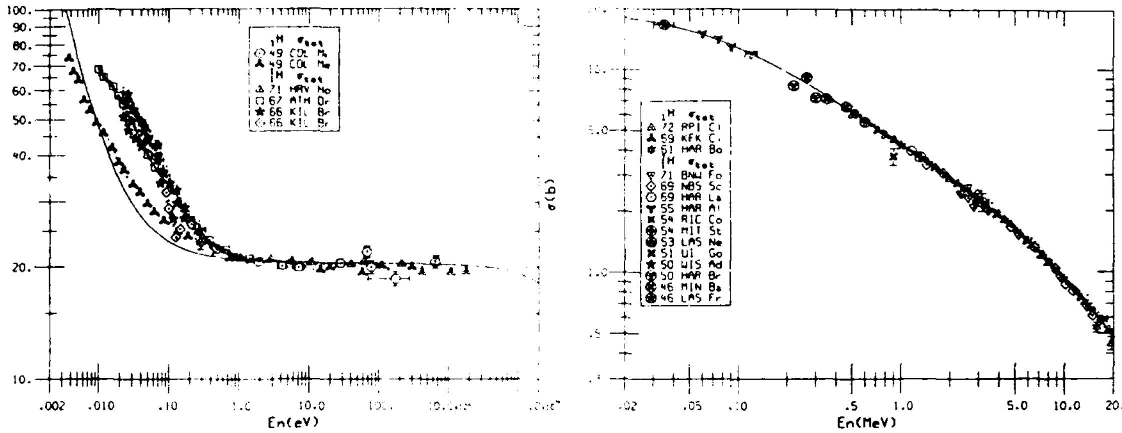

Figure 5, Total cross section of hydrogen. [From p. 1, Reference 2]

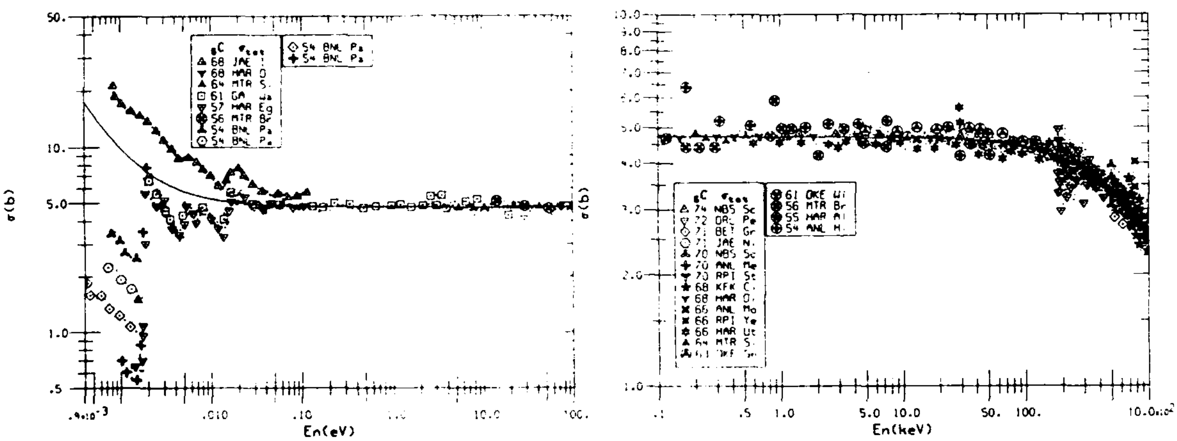

Figure 6, Total cross section of carbon. [From p. 28, Reference 2].

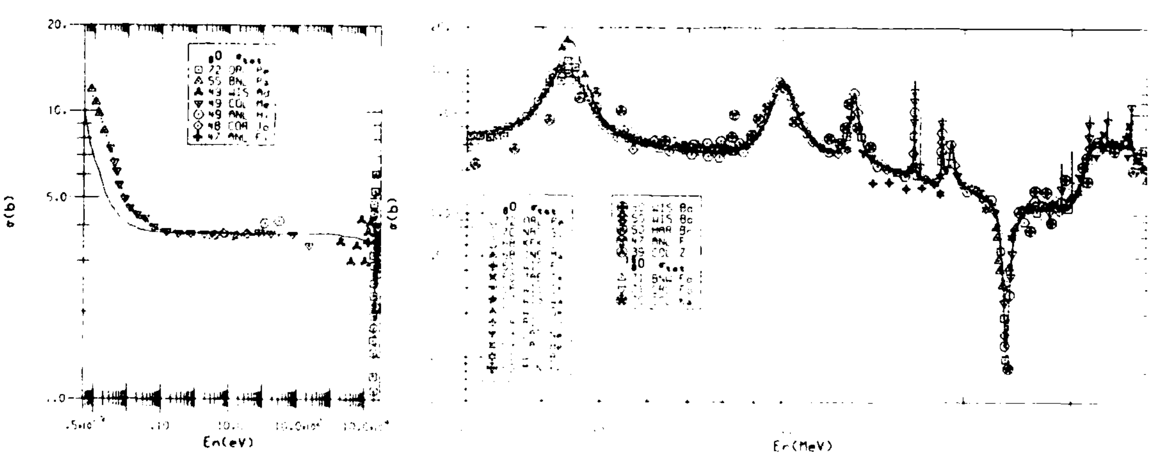

Figure 7, Total cross section of oxygen. [From p. 39, Reference 2].

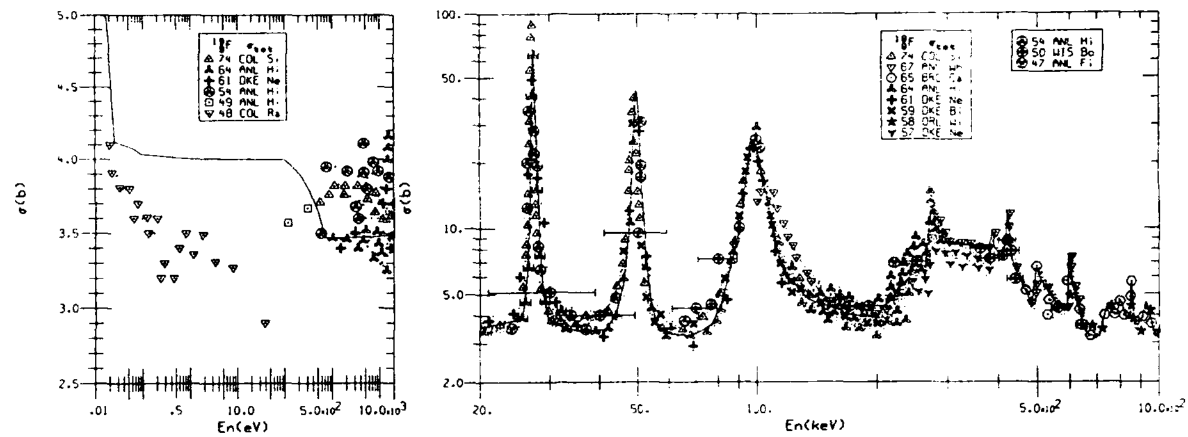

Figure 8, Total cross section of fluorine. [From p. 45, Reference 2]

With this energy cut off, the diffusion equation will simplify since

$$\chi_1 = \int_{1.0\,\mathrm{eV}}^{10\,\mathrm{MeV}} dE\chi(E) = 1.0\,, \tag{12}$$

$$\chi_2 = \int_{0\,\mathrm{eV}}^{1.0\,\mathrm{eV}} dE\chi(E) = 0\,. \tag{13}$$

Using equations 12 and 13 and applying equation 9 to two energy groups creates two simplified, coupled equations,

$$-\nabla \cdot D_1 \nabla \phi_1 + \Sigma_{R1}\phi_1 = \frac{1}{k}\left(\nu_1 \Sigma_{f1}\phi_1 + \nu_2 \Sigma_{f2}\phi_2\right)$$

$$-\nabla \cdot D_2 \nabla \phi_2 + \Sigma_{a2}\phi_2 = \Sigma_{s1\text{-}2}\phi_1\,. \tag{14}$$

Assuming that both the fast and thermal groups in a bare, uniform reactor have essentially the same spatial shape of $\Psi(r)$, where $\phi_1(r) = \Psi(r)\ \phi_1$ and $\phi_2(r) = \Psi(r)\ \phi_2$, we can show that

$$\nabla^2\psi(r)+B_g^2\psi(r) = 0\ , \tag{15}$$

where $\Psi(r_d)=0$ (r_d is the extrapolated radius) and $B_g{}^2$ is the geometric buckling. (Note that the subscript g in equation 15 does not denote an energy group, but rather the dependence of the buckling on system geometry.)

Then the relationships

$$\phi_1(r) = \phi_1\psi(r), \qquad \phi_2(r) = \phi_2\psi(r) \tag{16}$$

are substituted into Equation 14. Solving for k we find

$$K_{eff} = \frac{\nu_1\Sigma_{f1}}{\Sigma_{R1}+D_1B_g^2} + \frac{\Sigma_{s1\text{-}2}\nu_2\Sigma_{f2}}{\left(\Sigma_{R1}+D_1B_g^2\right)\left(\Sigma_{a2}+D_2B_g^2\right)}\ . \tag{17}$$

To determine the individual factors of equation 17 in terms of the two group constants, We will now draw an analogy between the two group solution for K_{eff} and the classical six-factor formula (equation 18), which provides the descriptors we want to describe thermalness. The classical six-factor formula is:

$$K_{eff} = \varepsilon\cdot\eta\cdot p\cdot f\cdot PNL_{fast}\cdot PNL_{thermal}\ . \tag{18}$$

The first step is to identify the diffusion length for each group as:

$$L_2^2 = \frac{D_2}{\Sigma_{a2}} \quad \text{(classical definition of } L_{thermal}^2\text{)}, \tag{19}$$

$$L_1^2 = \frac{D_1}{\Sigma_{R1}} \quad \text{(modified definition of } L_S^2\text{)}. \tag{20}$$

Using Equations 19 and 20, rearranging, and defining a fast multiplication factor, K_1, and a thermal multiplication factor, K_2, equation 17 becomes:

$$K_{eff} = K_1 + K_2 = \frac{\nu_1 \Sigma_{f1}}{\Sigma_{R1}\left(1+L_1^2 B_g^2\right)} + \frac{\Sigma_{s1\text{-}2}\nu_2\Sigma_{f2}}{\Sigma_{R1}\left(1+L_1^2B_g^2\right)\Sigma_{a2}\left(1+L_2^2B_g^2\right)}. \tag{21}$$

From basic one speed diffusion theory, the non-leakage probabilities are:

$$PNL_1 = \left(1+L_1^2B_g^2\right)^{-1}, \tag{22}$$

$$PNL_2 = \left(1+L_2^2B_g^2\right)^{-1}. \tag{23}$$

Considering the K_2 portion of equation 21 (the second term), we notice that the number of fission neutrons produced per thermal neutron absorbed is:

$$\eta_2 f_2 = \frac{\nu_2 \Sigma_{f2}}{\Sigma_{a2}}, \tag{24}$$

where η_2 and f_2 can be separated and identified as:

$$\eta_2 = \frac{\nu_2 \Sigma_{f2}^{FUEL}}{\Sigma_{a2}^{FUEL}}, \tag{25}$$

$$f_2 = \frac{\Sigma_{a2}^{FUEL}}{\Sigma_{a2}}, \tag{26}$$

which is consistent with the classical definitions. The remaining term in K_2 is $\Sigma_{s1-2}/\Sigma_{R1}$. This is just the number of neutrons slowing down into the thermal group per those removed from the fast group. This is identical to the definition of the resonance escape probability and therefore,

$$p = \frac{\Sigma_{s1-2}}{\Sigma_{R1}}. \tag{27}$$

Combining Equations 22 through 27 we get the thermal multiplication factor, K_2:

$$K_2 = \eta_2 \cdot f_2 \cdot p \cdot PNL_1 \cdot PNL_2. \tag{28}$$

Now considering the fast multiplication factor, K_1 and drawing an analogy to group 2 parameters, we can define a fast group $\eta_1 f_1$ as:

$$\eta_1 f_1 = \frac{\nu_1 \Sigma_{f1}}{\Sigma_{R1}}, \tag{29}$$

where the number of fission neutrons produced per fast neutron absorbed in fuel is:

$$\eta_1 = \frac{\nu_1 \Sigma_{f1}^{FUEL}}{\Sigma_{a1}^{FUEL}} \tag{30}$$

and the fast utilization factor is:

$$f_1 = \frac{\Sigma_{a1}^{FUEL}}{\Sigma_{R1}}. \tag{31}$$

Therefore, the K_1 term becomes:

$$K_1 = \eta_2 \cdot f_2 \cdot PNL_1 . \tag{32}$$

The only factor in the six-factor formula that has not been identified is the fast fission factor, ε. Knowing that:

$$K_{eff} = K_1 + K_2 , \tag{33}$$

and making use of the definition of K_{eff} by the six-factor formula (Equation 18), the fast fission factor can be expressed as:

$$\varepsilon = 1 + \left(\frac{\nu_1 \Sigma_{f1}}{\nu_2 \Sigma_{f2}}\right)\left(\frac{\Sigma_{a2} + D_2 B_g^2}{\Sigma_{s1-2}}\right). \tag{34}$$

With all of the factors of the six-factor formula defined in terms of the two-group constants, where:

$$K_{eff} = \eta_2 \cdot f_2 \cdot p \cdot \varepsilon \cdot PNL_1 \cdot PNL_2 \,, \qquad (35)$$

we can now evaluate systems with the two-group diffusion theory equation and define the new parameter for describing moderating capability.[1,2]

---

[1] This identification of the 2 group components of the six-factor formula is expanded from the treatment by Duderstadt and Hamilton, pp. 295-299, Reference 3.

[2] The equations for η_1, f_1, η_2, and f_2 are also derived for an appliction using a Fermi age lethargy 2 group cross section collapse analysis by Stanley, Reference 4.

2.2 Selection of the Parameter to Describe Moderating Capability.

In selecting the parameter to describe the moderating capability of a system, we can immediately eliminate some of the eight factors derived in the last section. The probabilities of non-leakage do not provide any explicit information about moderation so they can be eliminated. Similarly, η_1 and f_1, are fast group factors, which do not describe interactions resulting in moderation.

With the elimination of PNL_1, PNL_2, η_1, and f_1 this leaves η_2, f_2, ε, and p or some combination to be considered. To examine the behavior of these parameters over a range of moderator to fuel ratios, we modeled several critical $U(4.89)O_2F_2$, H_2O moderated and reflected homogeneous spheres (problems BRENT1 through BRENT4, Appendix 1). From this η_2, f_2, ε, and ρ were calculated and plotted against critical mass and volume. The results are shown in Figures 9 through 12. All four of these factors produce curves with the parabolic shape useful in predicting minimum critical mass and volume. Therefore, all four of these factors are good candidates for describing moderating capability.

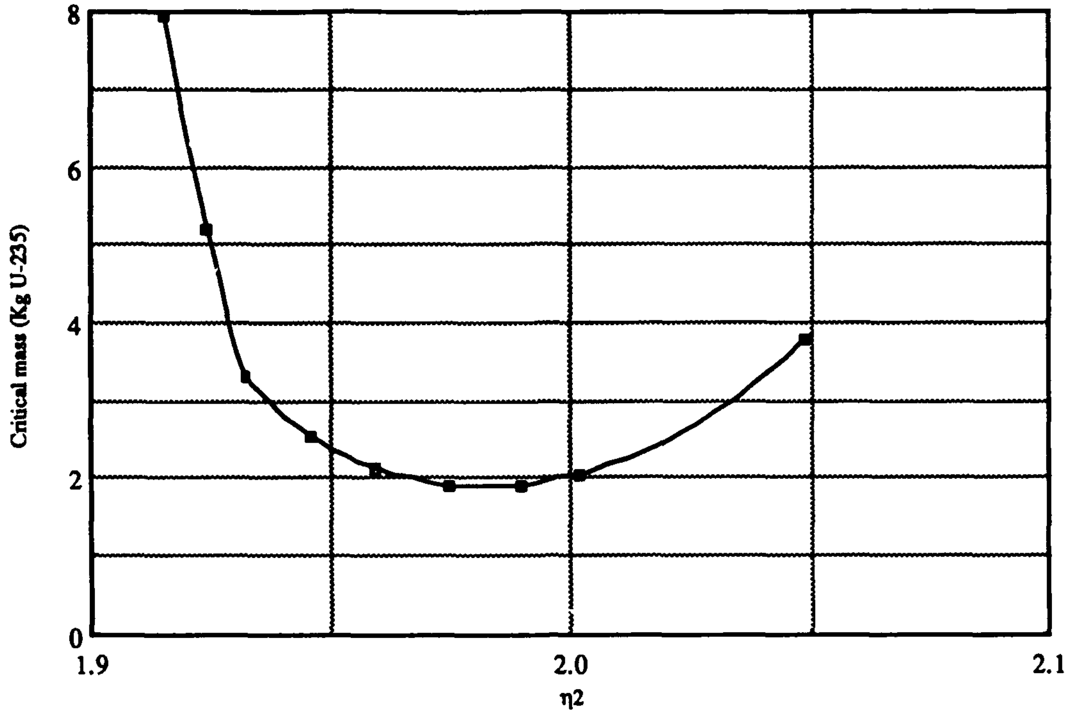

Figure 9, η_2 vs. critical mass for $U(4.89)O_2F_2$-H_2O, water reflected spheres of varying H/X.

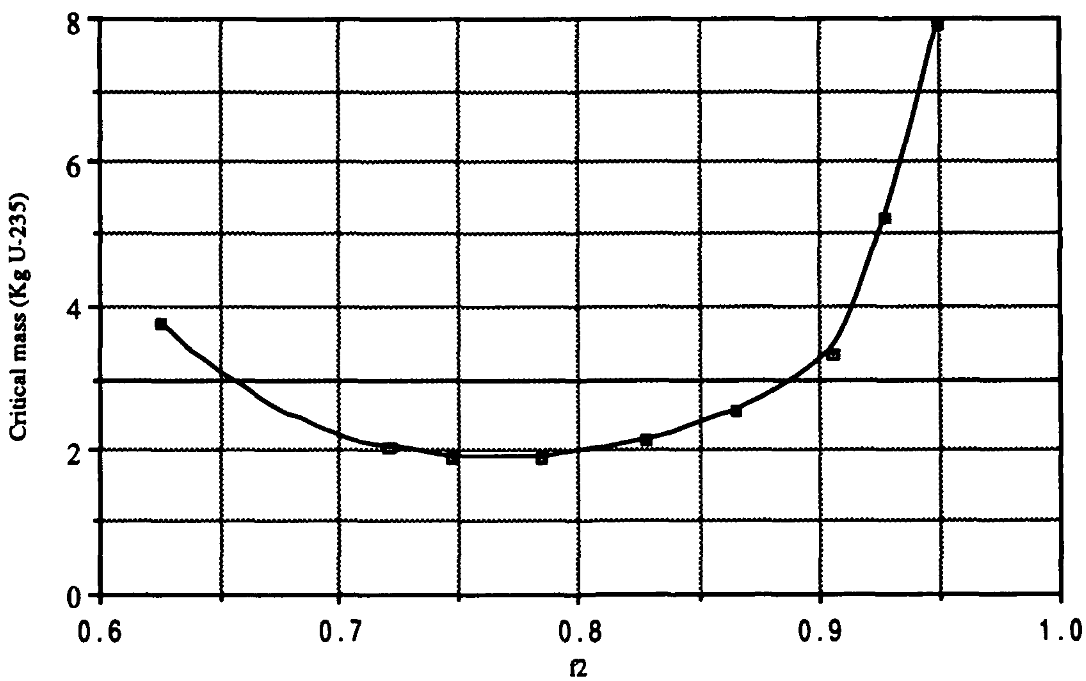

Figure 10, f_2 vs. critical mass for $U(4.89)O_2F_2$-H_2O, water reflected spheres of varying H/X.

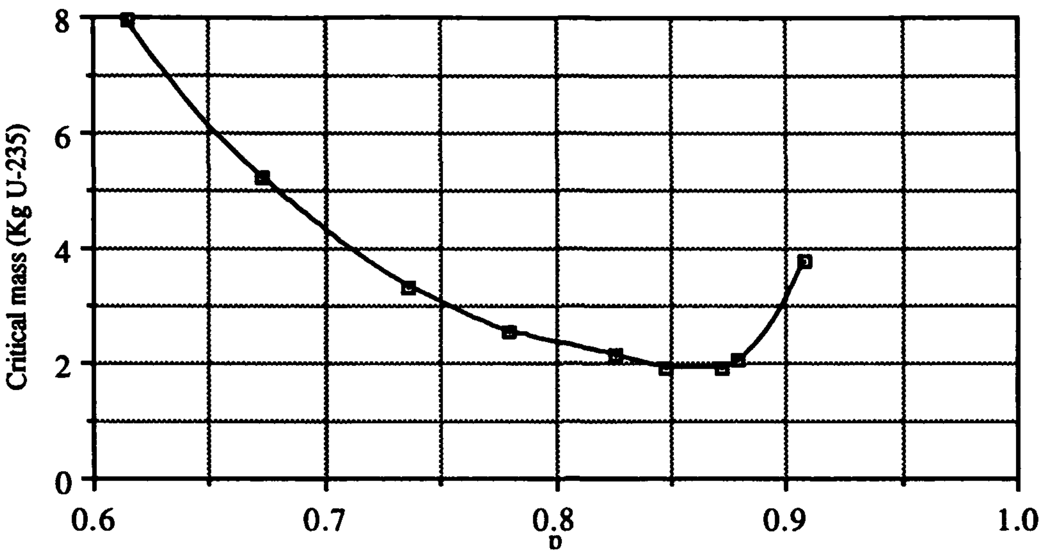

Figure 11, p vs. critical mass for U(4.89)O_2F_2-H_2O, water reflected spheres of varying H/X.

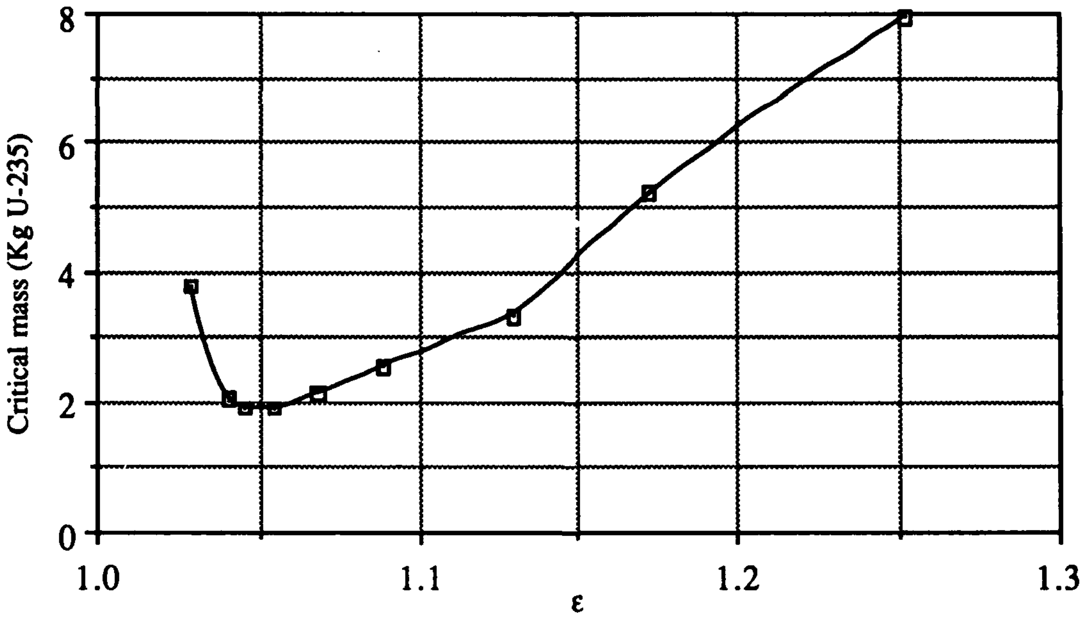

Figure 12, ε vs. critical mass for U(4.89)O_2F_2-H_2O, water reflected spheres of varying H/X.

The fast fission factor, ε, though representative of how "thermal" a system is, is not conducive for describing moderating capabilities. It gives a ratio of the number of neutrons from all fissions to the number of neutrons from thermal fissions, but does not say anything about the mechanisms of how the neutrons causing the thermal fissions were moderated. Additionally, the 2-group correlation for ε was derived by "backing in". In the relating of the two-group diffusion theory results to the six-factor formula, ε was the only term that did not readily present itself and we algebraically showed that, holding to the six-factor formula correlation, equation 34 had to be true. Another problem with ε is, in equation 34, it depends upon the buckling. Buckling is a factor derived from a one-speed diffusion theory approach for a bare, homogeneous reactor. It loses significance and meaning in multiregion or multigroup problems. Also, buckling is dependent on geometry and we are looking for a parameter that is geometry independent. (How we treat buckling in the modeling of critical systems will be discussed in Section 3).

The resonance escape probability, p, is an excellent measure of the slowing down characteristics of a system. It also treats the ability of the system to avoid resonance absorption. By the two-group relationship, equation 27, p considers all fast absorptions. The only thing p does not treat is how much thermal absorption there is in the system, but this is very important since an optimally moderated system will scatter while limiting undesirable absorptions. η_2 and f_2 are the measures of thermal absorption. Combined together they represent the number of neutrons produced from fission per thermal neutron absorbed. This is a relationship of desirable absorptions to undesirable absorptions. Therefore, a combination of η_2, f_2, and p should meet all the requirements of a parameter to describe the moderating capability of a system.

The most logical selection of the parameter would be a product of all 3 factors ($p^*\eta_2^*f_2$). This is in keeping with the form of the moderation ratio (equation 2), where it is the scattering term divided by the unwanted absorptive term. The product, $p^*\eta_2^*f_2$, by the definitions derived earlier, is the down scatter and desirable absorptions ($\nu\Sigma_f$) divided by the unwanted absorptions (Σ_{R1} and Σ_{a2}). This is analogous to the moderation ratio. However, when $p^*\eta_2^*f_2$ is plotted against critical mass (see Figure 13) we get an undesirable relationship. p varies inversly with $\eta_2 f_2$ as functions of fuel to moderator ratio. The product of the two then does not result in a very unique relationship to moderator to fuel ratio or critical mass and volume.

With the elimination of $p^*\eta_2^*f_2$ and keeping η_2 and f_2 together as one term, because of the computational convenience (alleviates the cumbersome definition of "fuel") and their likeness of physical meaning, a more suitable parameter is $p/(\eta_2^*f_2)$. As seen in Figure 14, this parameter behaves as we would like and encompasses all of the desired characteristics in describing moderation ability.

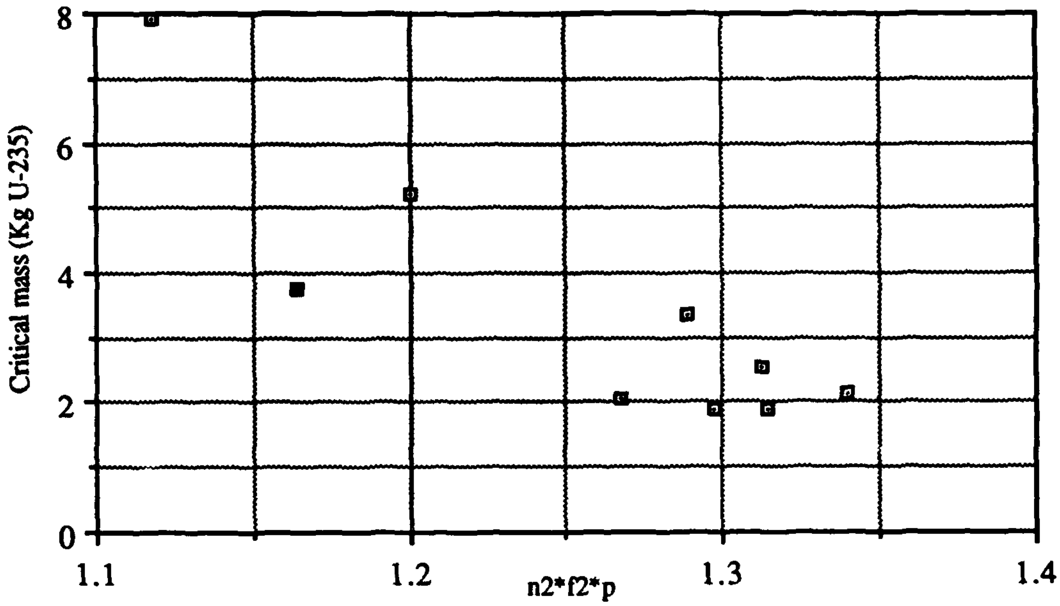

Figure 13, η_2*f_2*p vs. critical mass for U(4.89)O_2F_2-H_2O, water reflected spheres of varying H/X.

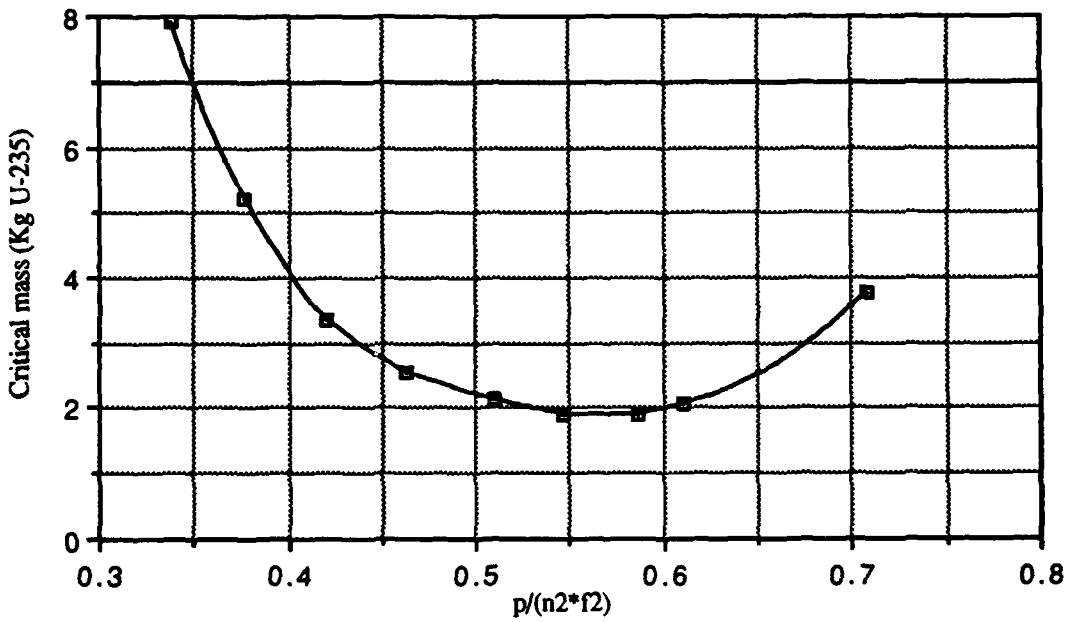

Figure 14, p/(η_2*f_2) vs. critical mass for U(4.89)O_2F_2-H_2O, water reflected spheres of varying H/X.

With $p/(\eta_2 {}^*f_2)$ selected as our parameter to describe moderating capability, we must now evaluate a selection of critical systems, calculate $p/(\eta_2 {}^*f_2)$, and evaluate the applicability of our parameter over a range of fuel-moderator mixtures and enrichments.

3. Modeling of Critical Systems In 2-Group Diffusion Theory

The critical systems evaluated are low enriched U^{235} systems with several different types of moderators. We chose low enriched systems as the focus of this study because they are more sensitive to moderator characteristics, specially the resonance escape characteristics. Table 1 specifies the fuel-moderator mixtures evaluated. Appendix 1, details each system modeled for which $p/(\eta_2{}^*f_2)$ was calculated. Wherever possible, actual data from critical experiments was used and modeled. When actual data was unavailable or inadequate for a desired fuel-moderator mixture, critical systems were calculated using the geometric eigenvalue search capabilities of the TWODANT code (References 5 and 6).

Table 1, Types of Critical Systems Evaluated

Fuel	Moderator	Enrichment (wt%)	Range of H/X
UO_2F_2	H_2O	4.89	100-1099
UO_2F_2	H_2O	30.3	76-815
U_3O_8	STEROTEX	4.89	102-449
UF_4	PARAFFIN	2.00	195-971

The first step in modeling the selected critical systems is to evaluate them with the TWODANT transport theory code and the Hansen-Roach 16-group cross section set (Reference 7). This is done for two reasons. It verifies that the system is correctly modeled, and it provides us with 16 group macroscopic cross sections for the system. These 16 group cross sections are then group-collapsed into two groups. Then using the two-group diffusion theory relationships developed in Section 2.1, the individual factors of the six-factor formula are calculated. K_{eff} is calculated using equation 35 and compared to the K_{eff} from the TWODANT 16-group cross section calculation to verify the two-group model. Finally $p/(\eta_2 * f_2)$ is calculated. The group collapsing and calculation of the six factors, K_{eff}, and $p/(\eta_2 * f_2)$ is done using a fortran program. (This program is presented in Appendix 2).

The only things left to consider are the treatment of the buckling term in the 2-group diffusion theory equations, which must be treated differently for bare and reflected systems, and the method of obtaining the two-group cross sections. The following sections deal with these very important topics.

3.1 Bare Sytems.

In computing K_{eff} using the two-group diffusion theory model derived in Section 2.1, we need to determine the buckling, B^2, of the systems. The first thing to consider is that buckling is an eigenvalue of the one speed diffusion theory equations. In the derivation of the two-group model, we asumed that both the fast and thermal fluxes had the same spatial shape or in other words the same buckling. Though some work has been done in expanding the meaning and use of buckling to two groups, we will continue to assume that buckling for our two group analysis can be represented by the one speed buckling of the systems.

For a bare system to be critical, the material buckling and geometric buckling must be equal. Since we are evaluating critical systems, we are able to use either the formula for material buckling,

$$B_m^2 = \frac{K_\infty - 1}{L^2} \tag{36}$$

or the formulas for geometric buckling,

$$B_g^2 = \begin{cases} \left(\frac{\pi}{a}\right)^2 + \left(\frac{\pi}{b}\right)^2 + \left(\frac{\pi}{c}\right)^2 \text{; parallelepiped} \\ \left(\frac{2.405}{R}\right)^2 + \left(\frac{\pi}{H}\right)^2 \text{; finite cylinder} \\ \left(\frac{\pi}{R}\right)^2 \text{; sphere} \end{cases} . \tag{37}$$

However, to use the material buckling we need a one group cross section set. As will be shown in Section 3.3, we introduce errors when developing a few group cross section set. The fewer the groups the less representative of the continuous energy dependent spectrum the cross sections become. Also, since we are group collapsing the Hansen-Roach 16 group cross section set to obtain our two group cross sections, there will be error associated with the collapse method (see Section 3.3). To further collapse these cross sections into one group creates additional error so we simply choose to use the the geometric relationships to calculate buckling.

To correctly use the geometric buckling relationships, we must calculate the extrapolation distance, d. For most cases d can be determined by

$$d = 0.71\lambda_{tr} = 2.13D \qquad (38)$$

This involves determining a one group diffusion coefficient. This is a trivial calculation and does not involve the collapsing and combination of as many parameters as the material buckling does.

3.2. Reflected Systems.

Recall that the six-factor formula and buckling are based on a bare, homogeneous reactor. In a reflected system we can no longer assume that the fast and thermal fluxes have the same spatial shape. To model reflected systems using the two-group diffusion theory analogy to the six-factor formula we make use of the concept of reflector savings. Reflector savings is the reduction in geometric dimensions of a critical system due to the addition of a reflector. For a critical sytem geometric buckling (equation 37) and material buckling (equation 36) must be equal. The addition of a reflector does not change the material buckling, therefore, the buckling of a reflected system is equal to the geometric buckling of an equivalent bare system. Making use of this relationship, buckling is determined for reflected systems in one of two ways.

The first method, and most preferred, is the use of an experimentally determined reflector savings. For many critical experiments the same fuel-moderator mixture was used in both a bare and reflected configuration. When equivalent bare dimensions are available, buckling for the reflected system is calculated by using the bare diminsions in equation 37. This is anologous to applying the reflector savings to a reflected system and treating it as an equivalent bare system.

When experimentally determined reflector savings are not available, we make use of a relationship derived from a coupled set of one group diffusion theory equations. The two equations are the one group diffusion equations for a spherical core and reflector. Solving these two equations, subject to boundry conditions, we find that:

$$BR\cot BR - 1 = -\frac{D_r}{D_c}\left(\frac{R}{L_r} + 1\right) . \tag{39}$$

The subscripts r and c refer to the reflector and core respectively. Given the radius, R, of the core, the buckling can be determined by an iterative approach [pp.214-217, Reference 8]. As in the buckling equations for a bare reactor we need the one group constants, D_r, D_c, and L_r. These are determined by group collapsing the Hansen-Roach 16-group cross section set as will be explained in the next section. Again we expect some error from the collapsing process. This relationship has one other limitation. It is only valid if $\tau_T << L_T^2$. Therefore, this relationship will not hold for reactors moderated and reflected by hydrogenous material where $\tau_T >> L_T^2$.

To handle hydrogenous moderated and reflected reactors we calculate the reflector savings, δ, using the emperical formula developed by R. W. Deutsch [p. 222, Reference 8],

$$\delta = 7.2 + 0.10\left(M_T^2 - 40.0\right), \tag{40}$$

to calculate the reflector savings. δ is added to the reflected system dimensions, converting it into an equivalent bare system, and buckling is determined as described in Section 3.1.

3.3. Group Collapsing of the Hansen-Roach 16-Group Cross Section Set.

The method described so far is dependent on identifying and using accurate 2-group neutron cross sections. There are a few 2-group cross section sets available. However, these are neither complete nor suitable for our purpose. One such 2-group cross section set is a neutron age collapse done by M. J. Stanely in 1958 (Reference 4). This set of cross sections is presented in macroscopic form based upon nominal densities. Not knowing for sure what he used as nominal densities, we are hesistant to use his values. It seems that with the introduction of multi-group computer codes using a variety of many group structured cross section sets, 2-group cross section sets have either not been developed or updated. Therefore, we have to develop a method for and determine the 2-group cross sections for use in this study.

This section explains the group collapse method we use in deriving 2-group neutron cross sections. We group collapse the Hansen-Roach 16-group cross section set into 2-groups by a flux weighting technique. There are questions concerning the accuracy of collapsing many group cross sections into fewer groups since we are becoming less and less descriptive of the continous cross section spectrum. Therefore, a great deal of effort has gone into attempting to quantify the extent of inaccuracy, where this inaccurracy occurs, and to justify the use of the collapsed cross sections in our 2-group analysis.

3.3.1. Definitions.

The following is a list of symbols and definitions introduced in this section.

Fine group = the many group structure prior to collapsing into broad groups. Denoted as g. See Figure 15.

Broad group = the few groups comprising of one or many fine groups after collapse. Denoted as G. See Figure 15.

si-i+n = scattering from group i to group i+n.

ICOL = A TWODANT defined input term for the edit energy-broad -group collapsing option. It is the number of fine groups in each broad group (example; ICOL=13,3 is 2 broad groups with 13 fine groups in the first, 3 in the second).

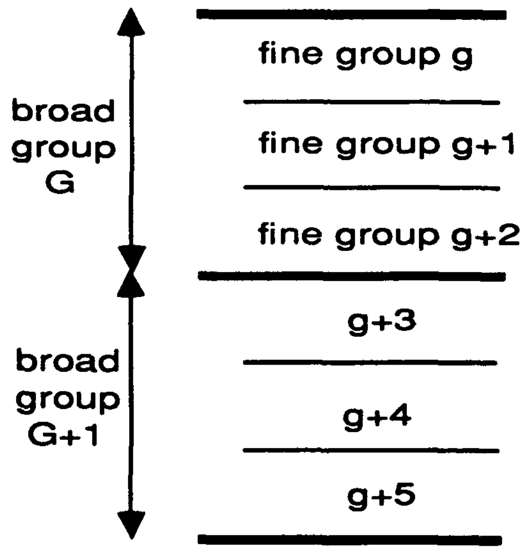

Figure 15, Example of fine and broad group structure.

3.3.2. Theory.

A. General.

The main principle in group collapsing is conservation of neutrons and neutron reactions. If these are conserved, there should be no difference in the ensuing calculation results.

The most common method of discretizing group constants is by performing two multigroup calculations. The first neglects any spatial or time dependence, and a finely structured multigroup calculation is done to determine the intragroup fluxes. This is done using appropriate models of neutron slowing down and thermalization. The group constants used in this first calculation are taken from tabulated continuous or many group cross section sets. The fluxes from the first calculation are then used to determine the broader group constants.

An example of this is taking the ENDF/B cross section data set and estimating the cross sections at a finite number of energy levels. The first approximation is normally just the average of the tabulated data over the energy range of each of the fine groups. This finite number of energy levels is larger than the number of groups to be used in the calculations. These are used to determine the "intragroup" fluxes that will be used in the final group collapse down to the desired number of broad groups for use in the calculations.

In a reactor or system, fluxes and cross sections are dependent on the composition and geometry of the system. Therefore, since group collapsing is based upon intragroup fluxes and the zone composition, the collapse method must take into account the space dependence of the flux and materials.

B. Methods of Group Collapsing.

The main method for collapsing cross sections into broader groups is by flux weighting. What this actually does is conserve the reaction rates of the finer groups within the broad group. The flux multiplied by the macroscopic cross section is the reaction rate. Summing the fine group reaction rates for a given broad group is the reaction rate for the broad group. Then dividing this by the total broad group flux (sum of the fine group fluxes within the broad group) gives the broad group macroscopic cross section. Equations 41 through 44 represent this for $v\Sigma_f$, the absorption and total cross sections, and the diffusion coefficient.

$$v\Sigma_{fG} = \frac{\sum_{i=g}^{g+n} \phi_i \times v\Sigma_{fi}}{\sum_{i=g}^{g+n} \phi_i} \tag{41}$$

$$\Sigma_{aG} = \frac{\sum_{i=g}^{g+n} \phi_i \times \Sigma_{ai}}{\sum_{i=g}^{g+n} \phi_i} \tag{42}$$

$$\Sigma_{tG} = \frac{\sum_{i=g}^{g+n} \phi_i \times \Sigma_{ti}}{\sum_{i=g}^{g+n} \phi_i} \tag{43}$$

$$D_G = \frac{\sum_{i=g}^{g+n} \phi_i \times D_i}{\sum_{i=g}^{g+n} \phi_i} \tag{44}$$

where the summation from i=g to g+n is over the fine groups contained within the broad group G.

Unfortunately, the process is not that straight forward for the scattering cross sections. Looking at Figure 16, we see that for each of the fine groups there are six scattering cross sections, $\Sigma_{sg\text{-}g}$ to $\Sigma_{sg\text{-}g+5}$ (Hansen-Roach 16-group cross section set structure). However, when collapsed into broad groups the self scatter and down scatter cross sections change based on the number of fine groups in each broad group.

For the case shown in Figure 16 the scattering cross sections for fine group 1, $\Sigma_{s1\text{-}1}$, $\Sigma_{s1\text{-}2}$, and $\Sigma_{s1\text{-}3}$, become self scattering cross sections for broad group 1. Similarly, the cross sections $\Sigma_{s2\text{-}2}$, $\Sigma_{s2\text{-}3}$, and $\Sigma_{s3\text{-}3}$ are also self scatterers for broad group 1. Then, corresponding with the treatment of the $\nu\Sigma_f$, Σ_a, and Σ_t collapsing, the broad group 1 self scatter cross section is

$$\Sigma_{sG\text{-}G} = \frac{(\Sigma_{s1\text{-}1}+\Sigma_{s1\text{-}2}+\Sigma_{s1\text{-}3})\phi_1+(\Sigma_{s2\text{-}2}+\Sigma_{s2\text{-}3})\phi_2+\Sigma_{s3\text{-}3}\phi_3}{\phi_1+\phi_2+\phi_3} \tag{45}$$

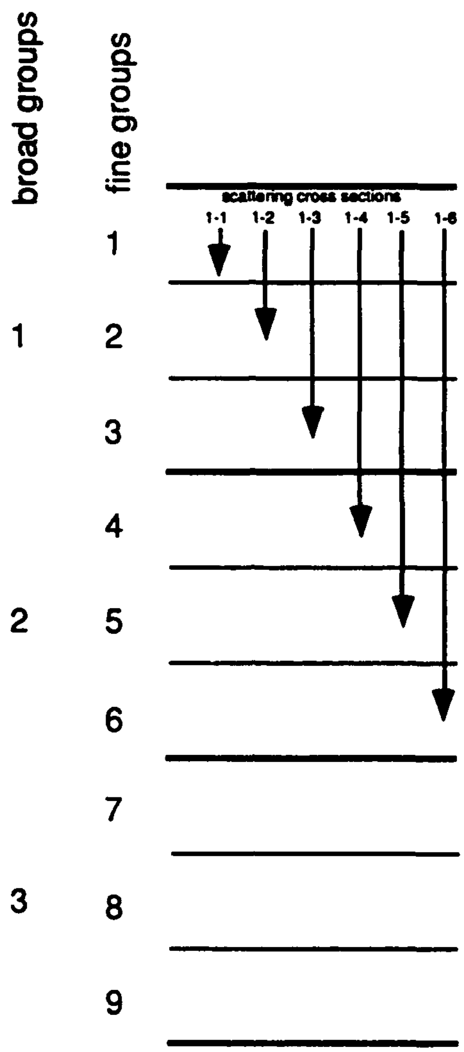

Figure 16, Relationship of the fine group 1 scattering cross sections and the broad group structure.

Broad group down scattering is similiarly evaluated. From the fine groups within a broad group, all the fine group scattering cross sections that result in a neutron energy within the bounds of specific lower broad group are componets of this broad group down scatter cross section.

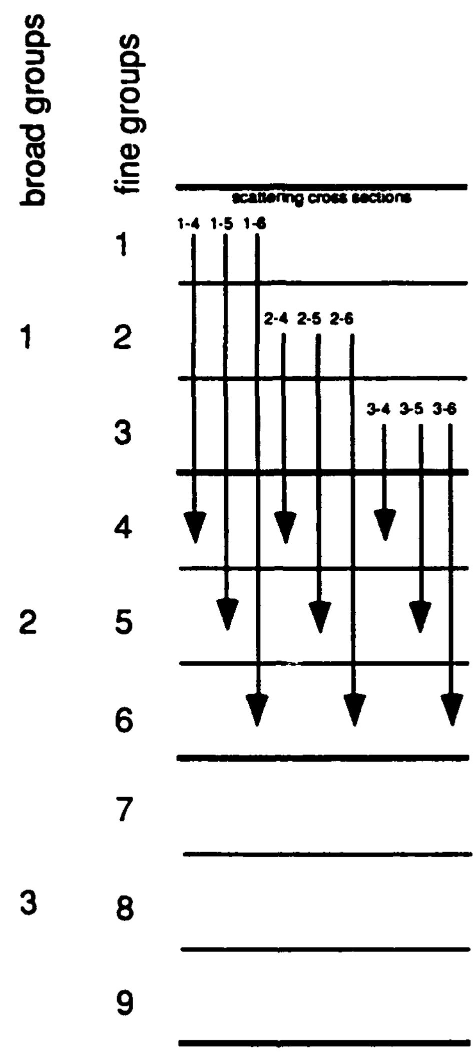

Figure 17, All fine group scattering cross sections that contribute to the broad group down scatter $\Sigma_{s1\text{-}2}$.

Figure 17 shows an example of all the fine group scattering cross sections in a broad group that are part of the scattering from this broad group down in to the next broad group. An evaluation of this nature must be done for each and every broad group and all possible combinations of broad group down scattering included.

C. Space Dependence.

Cross sections as defined and determined are a physical characteristic of a material or isotope. For a constant material in space (ie., a homogeneous mixture) there should be no variance in the cross sections for this material over space. Only for a change in material composition or density should a macroscopic cross section change with position. However, since cross sections are continuously energy dependent and we approximate this by multigroup techniques, we see some space dependence develop during the collapsing process. Note that we collapse by flux weighting; flux is space dependent. The ratios of the energy group fluxes also vary with position and, therefore, when collapsing cross sections this space dependence will become a factor.

The treatment of the space dependence falls into two categories: variations in material compositions and flux by position.

(1) Material Composition.

The treatment of the space dependence of material composition is fairly straight forward. For each zone of a specific material composition, a separate collapse is done to determine that zone's few group cross sections. As illustrated in Figure 18, a separate collapse needs to be done for the homogeneous core, the container, and the reflector to account for the space dependence of the material.

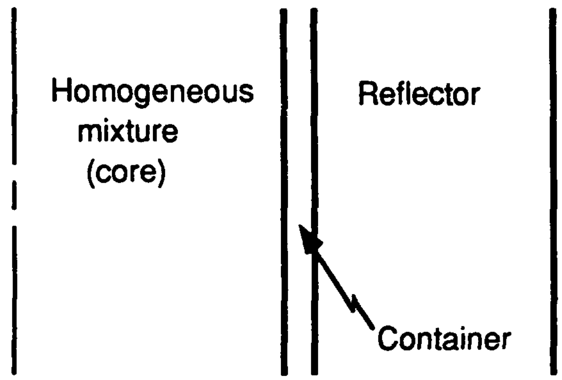

Figure 18, Mixture/composition variation with position.

(2) Flux.

The treatment of the space dependence of the flux when collapsing is slightly more complicated. We see in Figure 19, how the fluxes vary with position. Based on the use of flux weighting for group collapsing, it is obvious that our collapsed cross sections also vary with position. Discretizing space into mesh cells is an acceptable technique which reduces the number of collapsed cross sections. This reduces the continuous dependence into area or zone dependence. Of course the more zones or meshes used, the more accurate the outcome. (This will be further discussed in Section 3.3.4.A).

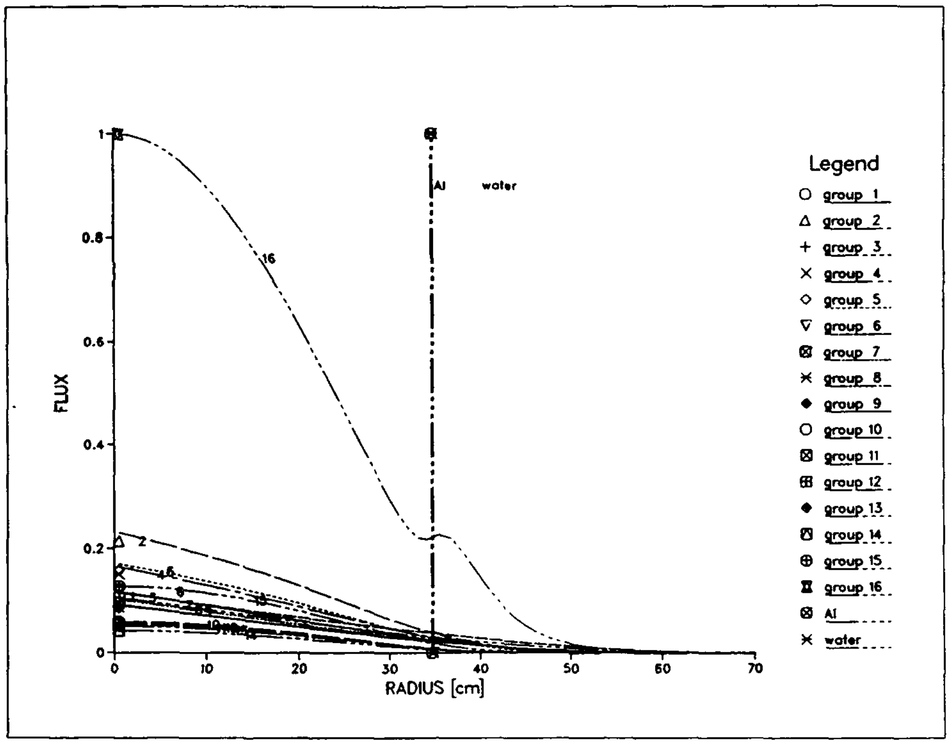

Figure 19, 16-group flux profile for a U(4.89)O_2F_2-H_2O, water reflected sphere with H/X=1099.

However, cross sections are physical characteristics and should not vary with position unless a material composition variance is encountered. The position variance of cross sections due to position variance of the flux is a result of the collapsing process. If a continuously varying cross section set is needed, we have done nothing to simplify our approach. It is unreasonable to have continuously varying solutions to our two-group diffusion theory calculations. Therefore, a further approximation is needed to account for this space dependence and obtain one set of cross sections for a given composition.

We can make an assumption that group collapsing with the average flux will provide the average cross sections. To determine the average flux, we further assume that one group diffusion theory is adequate to describe the flux profile of all the fine groups. For a sphere the one group diffusion solution for the flux is,

$$\phi(r) = \frac{1}{r}\sin\left(\frac{\pi r}{R}\right), \tag{46}$$

where R is the extrapolated radius for a bare sphere.

The average flux is determined by using the relationship,

$$\bar{\phi} = \frac{\int_V \phi(r)dV}{\int_V dV}, \tag{47}$$

where $dV = 4\pi r^2 dr$. Substituting equation 46 into equation 47, integrating and reducing we get:

$$\bar{\phi} = \frac{3}{\pi R}. \tag{48}$$

Equations 46 and 48 are combined to determine that the point where the average flux is found is,

$$r_{\bar{\phi}} = 0.6505R \tag{49}$$

This approximation uses the fine group fluxes at the point determined by equation 49, as a representative value of the entire system, to perform the group collapse.

The basic theory behind group collapsing is the conservation of reaction rates, $\phi * \Sigma$ [reactions/cm^3-sec]. However, the total number of reactions is volume dependent, $\phi * \Sigma * V$ [reactions/sec]. If we just use the average fluxes to collapse, we are assuming that the total reactions which occur are characteristic of the reaction rates at this point in space. This is not the case. The further from the center of a system the more volume a specific flux is related to.

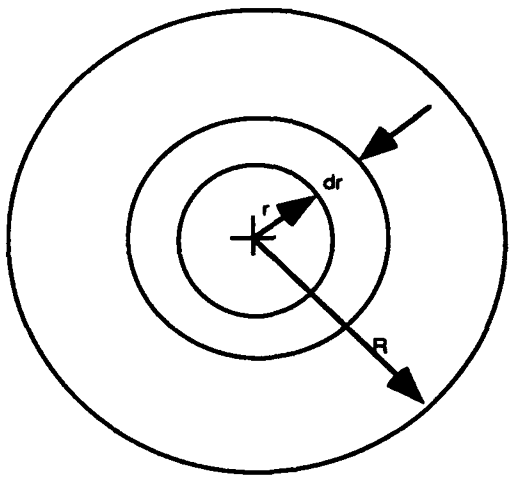

Figure 20, Differential volume of a sphere.

Referencing Figure 20, we note that the larger r, the more volume there is within the interval dr. Therefore, this volume effect on reaction rates can be used to further limit our collapse procedure. This is done by volume weighting the fluxes to obtain one representative flux for each group over the zone. This is the same as volume weighting the reaction rates by adding the volume terms to the numerators and denominators of equations 41-44 and all applicable scattering collapse equations. For example taking the absorption collapse relationship (equation 42) and volume weighting the reaction rates we get:

$$\Sigma_{aG} = \frac{\int^{v} \sum_{i=g}^{g+n} \phi_i \times \Sigma_{ai} dV}{\int^{v} \sum_{i=g}^{g+n} \phi_i dV}. \qquad (50)$$

However, only the flux and incremental volume are dependent upon position, so these can be separated from the fine group cross sections and we can separately volume weight the fluxes for each fine group. The volume weighted or average flux for each fine group is then,

$$\bar{\phi}_i = \frac{\int^{v} \phi_i(r) dV}{\int^{v} dV}. \qquad (51)$$

Discretizing this integral in space as deterministic codes do, we seperate our space into a finite number of cells. Using a one dimensional example, a sphere, when we descretize the space we have a set of concentric shells (Figure 21).

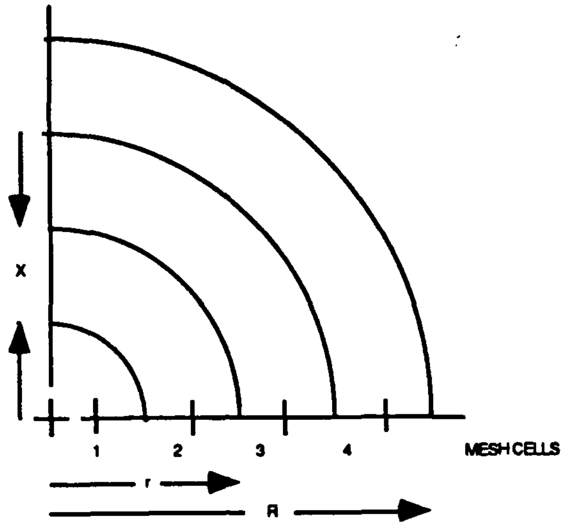

Figure 21, ¼ cut away of a sphere, radius R, descretized into 4 mesh cells of equal width x.

For each mesh cell we have a flux, ϕ_m, an internal radius r_m, and a mesh cell width of x (normally constant over a zone of the system). Rewriting equation 51 in terms of our discretizing technique we have,

$$\bar{\phi}_i = \frac{\sum_{m=1}^{M} \phi_{i_m} v_m}{\sum_{m=1}^{M} V_m}, \tag{52}$$

where M= the number of mesh cells used.

Using the example presented in Figure 21, we see that each mesh cell volume is determined by,

$$V_m = \frac{4}{3}\pi\left[(r+x)^3 - r^3\right], \tag{53}$$

and the denominator of equation 52 is just the total volume of the sphere ($^4/_3\pi R^3$). This whole process is nothing more than multiplying the fluxes by the volume over which they interact, summing these, and dividing by the total volume. This is easily generalized into other geometries. However, for two dimensional problems, both radial and axial directions need to be considered.

D. Angular Dependence/Anisotropic Effects.

Finite system fluxes and K_{eff} are heavily dependent upon the anisotropic scattering contributions and the angular flux or current. Looking at the steady state transport equation,

$$\begin{aligned}
&\nabla\cdot\vec{\Omega}\phi(r,E,\vec{\Omega}) + \Sigma_t(r,E)\phi(r,E,\vec{\Omega}) \\
&\quad = \int dE'\int d\Omega'\Sigma_s(r,E'\to E,\vec{\Omega}'\to\vec{\Omega})\phi(r,E,\vec{\Omega}) \\
&\qquad + \frac{1}{4\pi}\int dE'\int d\Omega'\chi(r,E'\to E)\nu\Sigma_f(r,E')\phi(r,E,\vec{\Omega}) \\
&\qquad + Q(r,E,\vec{\Omega})\,,
\end{aligned} \tag{54}$$

we immediately notice that the angular dependence is on the right hand side of the equation. The left hand side, leakage and removal terms, is not angular dependent. If a neutron leaks from the incremental volume or interacts in any method $[\Sigma_t\phi(r,E,\Omega)]$, regardless of neutron incident or final direction, it is a loss in terms of the balance equation. On the right hand side of the equation, the fission source term is assumed to be isotropic. The fission neutrons have basically forgotton the direction of the incident neutron and are ,therefore, independent of this direction. We will neglect the flux independent source term, $Q(r,E,\Omega)$, since it is not a factor in this investigation.

This leaves the inscatter source term,

$$\int dE' \int d\Omega' \Sigma_s(r,E' \to E,\vec{\Omega}' \to \vec{\Omega}) \phi(r,E,\vec{\Omega}), \tag{55}$$

which can be highly angular dependent. The angular dependence is commonly treated by use of a spherical harmonics expansion and/or discrete ordinates methods. In the spherical harmonics expansion technique the scattering transfer propability, $\Sigma_s(r,E'-E,\Omega'-\Omega)$, is represented by a Legendre polynominal expansion:

$$\Sigma_s(r,E' \to E,\vec{\Omega}' \to \vec{\Omega}) = \sum_{n=0}^{L} \left(\frac{2n+1}{4\pi}\right) \Sigma_{sn}(r,E' \to E) P_n(\mu_0), \tag{56}$$

where $\mu_0 = \Omega' \bullet \Omega$. The $\Sigma_{sn}(r,E'-E)$ term is the anisotropic scattering cross section. The spherical harmonics theorem is used to expand $P_n(\mu_0)$ to further treat the angular dependence and the in-scatter term becomes:

$$\int_{E} dE' \sum_{n=0}^{L} \left(\frac{2n+1}{4\pi}\right) \Sigma_{sn}(r,E' \rightarrow E)$$

$$\left\{ P_n(\mu) \int_{-1}^{1} d\mu' \int_{0}^{2\pi} d\theta' p_n(\mu')\phi(r,E',\mu',\theta') + 2\sum_{k=1}^{n} \frac{(n-k)!}{(n+k)!} P_n^k(\mu) \int_{-1}^{1} \int_{0}^{2\pi} d\theta' P_n^k(\mu') \cos k(\theta-\theta')\phi(r,E',\mu',\theta,) \right\}. \tag{57}$$

Notice that even though the scattering cross sections angular dependence is now in terms of an energy only dependent anisotropic scattering cross section, the angular dependence is still present in the expansion of $P_n(\mu_0)$.

The discrete ordinates method also involves the Legendre polynominal expansion of the scattering transfer probability. The angular dependence, however, is treated by characterizing the angular-direction domain as a finite number of quadrature points each with an associated quadrature weight. Now we have the inscatter term as

$$\int_{E'} dE' \sum_{n=0}^{L} \sum_{m=-n}^{n} \Sigma_{sn}(r,E' \rightarrow E) \left[\sum_{d'=1}^{D} \omega_{d'} \overline{Y}_n^m(\Omega_{d'}) \phi(r,\Omega_{d'},E') \right] Y_n^m(\Omega_d), \tag{58}$$

where d=1,2,. . .,D, the quadrature directions, ω_d is the quadrature weight, $Y^m{}_n$ is the spherical harmonic relationship, and $Y^m{}_n$ is the complex conjugate of $Y^m{}_n$. Again the angular dependence is found in the expression operating on the anisotropic scattering cross section.

Our Hansen-Roach cross section sets have anisotropic scattering cross sections associated with hydrogen and deuterium (the two moderators with a significant anisotropic component). These can be group collapsed as discussed earlier. The concern is with the treatment of the angular dependence as found in the spherical harmonics and descrete ordinates approaches (i.e., the functions operating on $\Sigma_{sn}(r,E'-E)$ in equations 57 and 58). To reproduce the exact results of a calculation with a group collapsed set of cross sections, the angular dependence of the scattering must be preserved. This cannot be done using scalar fluxes. Since the angular-directional fluxes of each fine group determine the scattering process. But it is not feasible to collapse at each point, for each angular flux direction as the resultant cross section set would be to complicated for practical use in our simplified two-group analysis.

3.3.3. General Method of Analysis/Procedure.

A. Basic Procedure.

The tools used in evaluating errors associated with group collapsing are the deterministic transport theory code, TWODANT, with the Hansen-Roach (H-R) 16-group cross section set, BXSLIB, and a simple fortran program, COLLAPSE, which group collapes cross sections by flux weighting as outlined in the theory section. (This program is detailed in Appendix 3).

The basic procedure followed in this investigation begins with modeling a system using TWODANT/H-R. TWODANT's output contains the macroscopic cross sections and flux profile for each of the sixteen groups (References 5 and 6). Corresponding with the earlier discussion on the basic theory of collapsing, the Hansen-Roach cross sections are the 16 fine groups. The TWODANT derived 16-group fluxes are the intragroup fluxes. Once the system is accurately modeled in TWODANT, the macroscopic cross sections and flux profiles are used as input to COLLAPSE to determine the group collapsed cross sections for a desired set of broad groups. Volume weighting of the flux to determine a volume average flux is done using a simple fortran program, FLUX (Appendix 4), which applies the theory discussed in Section 3.3.C.(2). If volume weighted fluxes are desired for the group collapse, the output from FLUX is used as the input fluxes for COLLAPSE. Then the group collapsed cross sections are entered into TWODANT (Reference 5), and holding everything else constant, the resultant multiplication factor is compared to the one derived using the H-R cross section set.

B. Evaluation of Sources of Errors.

We approached the evaluation of the errors associated with the group collapsing of cross sections in 3 steps. These were the evaluation of the fluxes used for the collapse, collapsing with different ICOLs in different cross section regions (thermal, epithermal, resonance, and fast), and the evaluation of the anisotropic/P1 effect.

(1) Fluxes.

The first question is how significant is the choice of the fluxes used for the group collapse. Is there a significant difference in using the volume weighted fluxes versus the centerline or "average" fluxes in deriving one collapsed cross section set for a zone? To answer this group collapses were done using centerline, average, and volume weighted average fluxes for the same system and ICOL. Additionally, these were compared to point by point collapses, where a collapse was done at each TWODANT mesh point and then modeled in TWODANT as separate zones with a separate set of group collapsed cross sections for each zone.

(2) Varying ICOLs.

To determine if there is a specific energy range in the cross section spectrum that produces the majority of the error when group collapsing is done, we group collapsed over a wide range of ICOLs. The H-R 16-group cross section set is broken down as shown in Table 2.

Table 2, Hansen and Roach 16-Group Cross Section Specifications
[extracted from p.12, Reference 7].

Group	Energy Range	U^i	v, cm/shake	Fission Spectrum
1	3 - ∞ Mev		28.5	0.204
2	1.4 - 3 Mev	0.762	19.9	0.344
3	0.9 - 1.4 Mev	0.442	14.7	0.168
4	0.4 - 0.9 Mev	0.811	11.0	0.180
5	0.1 - 0.4 Mev	1.386	6.7	0.090
6	17 - 100 Kev	1.772	2.70	0.014
7	3 - 17 Kev	1.735	1.14	0
8	0.55 - 3 Kev	1.696	0.480	0
9	100 - 550 ev	1.705	0.206	0
10	30 - 100 ev	1.204	0.101	0
11	10 - 30 ev	1.099	0.0566	0
12	3 - 10 ev	1.204	0.0319	0
13	1 - 3 ev	1.099	0.0179	0
14	0.4 - 1 ev	0.916	0.0109	0
15	0.1 - 0.4 ev	1.386	0.00606	0
16	Thermal (0.025)		0.00218	0

[where U^i is the Lethargy width of group i]

Since the emphasis of the study is on uranium systems with significant resonance absorption, we used the energy dependent total cross sections for U^{235} and U^{238} as a reference (see Figures 3 and 4), and we grouped the H-R cross section energy groups into four broad energy groups. The fast region, governed by the fission spectrum, is groups 1 through 6. The resonance region is the energy range below the fast region and ending in the last resonance of the uranium isotopes, groups 7 through 12. The thermal region, based on our two-group diffusion theory objective, is groups 14 through 16. This leaves group 13 unclassified. We will designate it as the lone epithermal group.

Using these definitions, collapses were done starting with the combination of just two of the H-R groups within a defined region, then three, until the entire region was collapsed into one broad group. This was done leaving all the other H-R groups as is to isolate the effect of the collapse to the specific region of interest. Other combinations, collapses across region boundries and selected ICOLs, were also conducted to evaluate the effect these had on the accuracy of the multiplication factor, K. These were all done using the volume weighted average fluxes; the more accurate method of developing one cross section set per zone (this will be detailed in the results of the analysis).

(3) Anisotropic/P1 Effect.

The first angular dependent check was to run K_{∞} calculations. In an infinite system the fine group fluxes are constant and the scattering is truly isotropic. Therefore, K_{∞} calculations and collapses are completely independent of angle or direction. The other check was to run K_{eff} calculations with the Legendre expansion order equal to 0. However, TWODANT still assigns quadrature directions and weights for this case so angular dependence is still present.

C. *Important Points When Group Collapsing Using TWODANT/H-R for the Fine Group Structure.*

(1) The anisotropic component of the H-R cross section set must also be group collapsed and included in the input of the group collapsed cross sections, unless the Legendre expansion order is 0.

(2) TWODANT determines the effective absorption cross section by subtracting the scattering cross sections from the total cross section. *The total cross section and scattering cross* sections are normally on the order of 10^{-1} or 1. The absorption cross sections are normally on the order of 10^{-3} or 10^{-2}. If not enough significant digits are included in the input cross sections, when TWODANT computes the absorption cross section, the absorption cross section used in the calculation can be significantly different from the one derived by the group collapse method. SIGNIFICANT DIGITS ARE SIGNIFICANT.

(3) *In determining the group collapsed diffusion coefficients,* you must first determine the fine group diffusion coefficients and then group collapse these. Using the group collapsed cross sections to determine a broad group diffusion coefficient does not result in

the correct value. The equation for determinig the fine group diffusion coefficients, consistent with the TWODANT code, is:

$$D_g = \frac{1}{3(\Sigma_{t_g} - \Sigma_{s1_{g \to g}})} . \tag{59}$$

3.3.4. Results/Analysis.

A. Fluxes Used for the Group Collapse.

The first thing considered is how does the choice of fluxes used in the group collapse impact the results. Group collapses of ICOL=13,3 (ICOL used in two-group diffusion theory analysis) were done on two $U(4.89)O_2F_2\text{-}H_2O$, bare and reflected systems, a U(93)-C system, and a U(93) metal system. For each system, collapses were done using the center point flux, average flux as determined by equation 48, and the volume weighted average flux (VA flux). The resulting two group cross sections were provided as input to TWODANT and, with everything else constant, K_{eff} was calculated.

To further evaluate flux dependence on the collapse procedure, the systems were modeled in TWODANT with as few mesh points as possible, and a collapse was done at each mesh point with that point's fine group fluxes (space/energy collapse). Then K_{eff} was calculated with these input into TWODANT as separate zones, each with a unique set of cross sections. This should eliminate any approximations made in selecting one flux that is representative of the entire material zone. The results of these calculations are shown in Table 3.

Table 3, Comparison of K_{eff} from group collapses using different fluxes.

		K_{eff}			
		group collapsed with ICOL=13,3			
	H-R 16 group	by the following fluxes:			
Problem[a]	cross sections	centerline	average	volume weighted	space/energy
BRENT1	1.03878	1.06699	b	1.06608	1.06303[c]
BRENT2	1.01521	1.07729	b	1.07540	1.07491[d]
BRENT10	0.984257	1.03699	1.03562	1.03201	1.03365[e]
BRENT11	0.992137	1.04059	1.04005	1.03717	1.03820[f]
BRENT50	0.999964	1.05551	1.03276	1.04900	1.04922[g]
BRENT60	1.0002	1.01018	1.00905	1.00830	1.00755[g]

a. All problems ran with the Legendre polynominal order of scattering as 1.
b. Reflected systems. Equation 8 is invalid.
c. 25 mesh points; 12 in core, 1 in Al container, 12 in reflector.
d. 20 mesh points, 10 in core, 1 in Al container, 9 in reflector. For value shown, 2 mesh points per collapsed mesh point were used (see explanation in write up).
e. 13 mesh points, 12 in the core, 1 in the Al container.
f. 11 mesh points,10 in the core, 1 in the Al container.
g. 10 mesh points.

What is obvious here is that in most cases the space/energy collapse doesn't provide the closest approximation to the H-R 16 group solution. The volume weighted average flux is the most accurate. One thing to note is that in doing the space/energy collapses, we have induced some error by reducing the number of mesh points used by TWODANT to reduce the number of calculations.

Looking at problem BRENT2 as an example, the problem was run with 10 mesh points in the fuel bearing region, 1 in the Al container, 8 in the first 17 cm of the reflector, and 4 in the remaining 90 cm of reflector. The breakdown in the reflector was neccessary to adequately model the drastically varying flux profiles in the "thermal hump" region of the reflector. Also, the collapses in the outer region of the reflector became meaningless due to fluxes on the magnitude of 10^{-40}, with the group 16 flux still on a magnitude of 10^{-16}. The collapse fell apart and reduced to values of zeroes because of the computers inability to handle these small numbers. It was determined that neglecting the last 30 cm of the reflector didn't alter the results of K_{eff} when using the space/energy collapsed cross sections.

What does alter the value of K_{eff} is the number of fine mesh points used in the calculation. The BRENT2 $K_{eff,}$ using the H-R 16-group cross sections, shown in Table 3 was calculated using 117 mesh points, 40 in the core, 2 in the Al container, and 75 in the water reflector. K_{eff} was calculated as 1.01528 using the mesh point break down described in the preceding paragraph and the H-R 16-group cross sections. Though an extremely small difference in K_{eff}, there is a variation in the flux profiles of the two calculations (Figure 22 and 23). From Figures 22 and 23 we see that the largest difference in the flux profiles is in the 16th group near the material/reflector boundry. This variation in flux is the reason we see that space/energy collapsed cross sections, often produce K_{eff}s with more error than the volume averaged flux method. Also, when the space/energy collapsed cross sections were entered into TWODANT as separate zones, using only one mesh point per zone, K_{eff} was 1.07553 compared to a K_{eff} of 1.07491 using two mesh points per zone. Table 4 provides K_{eff} values for additional cases with various numbers of mesh points and energy groups.

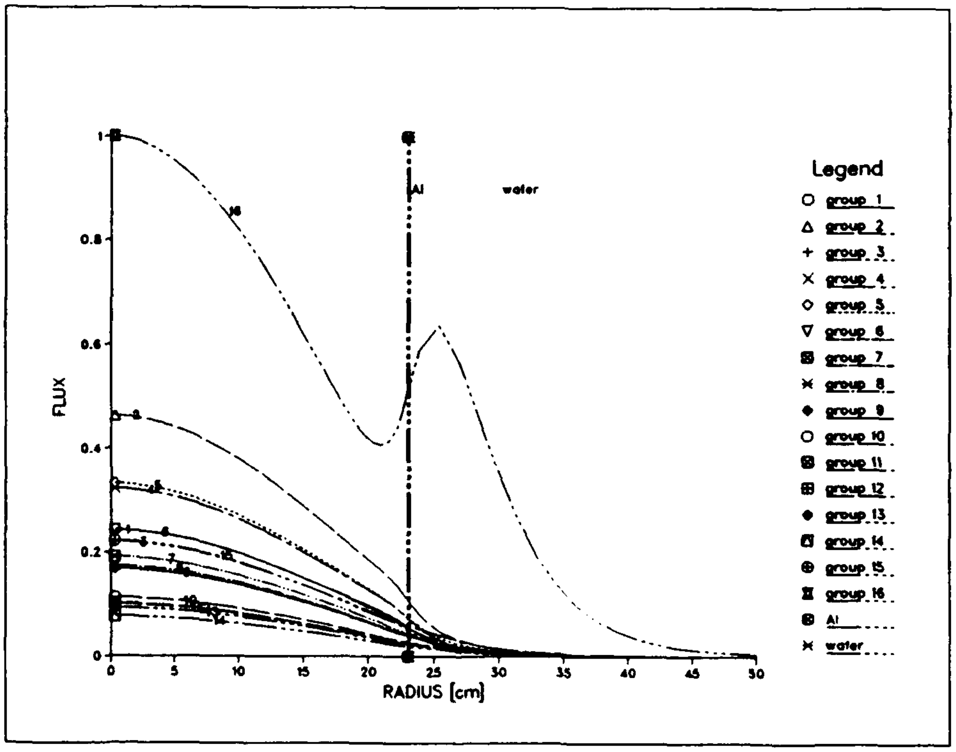

Figure 22, BRENT2, U(4.89%)O_2F_2-H_2O, water reflected sphere, H/X=524, TWODANT derived flux profile with XINTS=40,2,75.

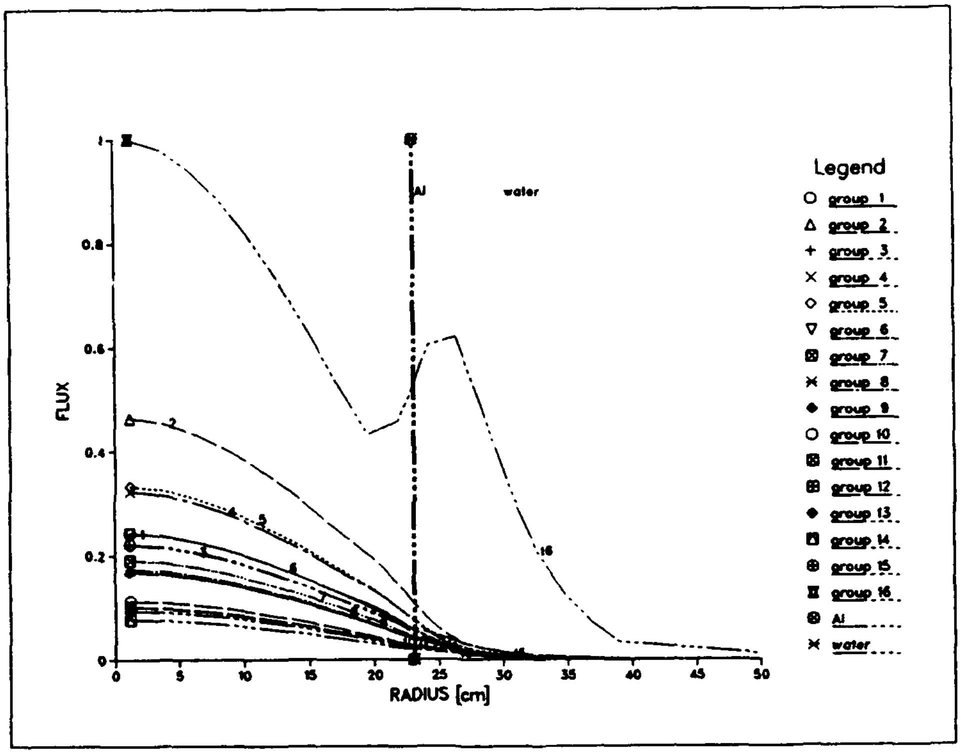

Figure 23, BRENT2, U(4.89%)O_2F_2-H_2O, water reflected sphere, H/X=524, TWODANT derived flux profile with XINTS=10,1,8,4.

Table 4, Comparison of results for different numbers of mesh points used in TWODANT K_{eff} calculations.

Problem	Number of energy groups[a]	XINTS[b]	Mesh Size per zone (cm)[b]	TWODANT determined K_{eff}
BRENT1	16	34,2,75	1.02,0.08,1.36	1.03878
	16	12,1,12	2.89,0.16,8.53	1.03727
	2	12,1,12	2.89,0.16,8.53	1.06303
	2	24,2,24	1.44,0.08,4.26	1.06704
BRENT2	16	40,2,75	0.58,0.08,1.52	1.06704
	16	10,1,8,4[c]	2.3,0.16,2.11,24.3	1.01528[d]
	2	10,1,8,4	2.3,0.16,2.11,24.3	1.07553
	2	20,2,16,8	1.15,0.16,1.05,12.1	1.07491
BRENT10	16	75,5	0.339,0.03	0.984257
	16	12,1	2.12,0.16	0.983033
	2	12,1	2.12,0.16	1.04922
BRENT11	16	40,2	0.71,0.08	0.992137
	16	10,1	2.82,0.16	0.991940
	2	10,1	2.82,0.16	1.03820
	2	20,2	1.41,0.16	1.03826
	2	30,3	0.94,0.05	1.03827
BRENT50	16	60	1.09	0.999964
	16	10	6.56	0.995230
	2	10	6.56	1.04922
	2	20	3.28	1.04946
BRENT60	16	10	0.87	1.00018
	2	10	0.87	1.00755

a. All two group calculations presented used the space/energy collapsed cross sections from the smallest number of 16 group XINTS listed for the problems.

b. XINTS is the TWODANT code for the number of mesh cells per zone. They are presented here in the order: core/fuel bearing region,container,reflector.

c. This break down of XINTS has two separate zones in the reflector.

d. This would not fully converge. The discrepancy was in the last mesh cell in the reflector.

The results shown in Table 4 confirm that the number of mesh points used in the collapse calculation, and in the modeling of the systems, have an impact on the accuracy of the K_{eff}. The deviation seems to follow no set pattern and is system dependent. To accurately model and collapse using the space/energy method we must use an appropriate number of mesh points, which is usually large. This involves a large number of calculations and produces a large number of cross sections needed to describe one system. This is inappropriate for our needs.

For our simple two group diffusion theory approach, we only want one set of cross sections for a given material. With this in mind, the results presented in this section show the volume weighted average flux provides the best results.

The concern with these results is the apparent magnitude of the error when group collapsing with any of the methods presented above. What we will do now is try to isolate what region of the energy spectrum the majority of this error comes from and why.

B. Group collapses of varying ICOLs.

Using the volume weighted average fluxes for collapsing, two systems were group collapsed into the combinations of energy groups and ICOLs presented in Tables 5 and 6. As done in evaluating the choice of fluxes, the collapsed cross sections were provided as input into TWODANT and K_{eff} calculated.

Table 5, Results from collapsing the H-R 16-group cross section set for a U(4.89%)O_2F_2-H_2O, water reflected sphere with H/X=1099.

#of groups	cross sections/ICOL	Keff	|%Δ|	K_∞	|%Δ|	remarks
16	H-R 16 group set	1.03878	0.00	1.19684	0.00	
16	16 groups/manual input	1.03892	0.01	1.19701	0.01	
15	2,1,1,1,...1	1.03915	0.04	1.19879	0.16	fast group collapse [fission spectrum; groups 1-6]
14	3,1,1,1,...1	1.04071	0.19	1.19639	0.04	
13	4,1,1,1,...1	1.04337	0.44	1.19919	0.19	
12	5,1,1,1,...1	1.04736	0.85	1.19626	0.05	
11	6,1,1,1,...1	1.05145	1.22	1.19901	0.18	
15	1,1,1,...1,2	1.04219	0.33	1.19593	0.08	thermal/epithermal collapse [groups 13-16]
14	1,1,1,...1,3	1.04137	0.25	1.19673	0.01	
13	1,1,1,...1,4	1.04187	0.30	1.19540	0.12	
15	1,1,1,1,1,1,1,1,1,1,2,1,1,1,1	1.03895	0.02	1.19703	0.02	resonance region collapse [groups 7-12]
14	1,1,1,1,1,1,1,1,1,3,1,1,1,1	1.03901	0.02	1.19702	0.02	
13	1,1,1,1,1,1,1,1,4,1,1,1,1	1.03907	0.03	1.19701	0.01	
12	1,1,1,1,1,1,1,5,1,1,1,1	1.03919	0.04	1.19710	0.02	
11	1,1,1,1,1,1,6,1,1,1,1	1.03928	0.05	1.19715	0.03	
10	1,1,1,1,1,7,1,1,1,1	1.03896	0.02	1.19640	0.04	resonance region collapse expanded into fast region
9	1,1,1,1,8,1,1,1,1	1.04083	0.20	1.19697	0.01	
8	1,1,1,9,1,1,1,1	1.04331	0.44	1.19682	.002	
7	1,1,10,1,1,1,1	1.04576	0.67	1.19671	0.01	
6	1,11,1,1,1,1	1.05615	1.67	1.19685	0.00	
5	12,1,1,1,1	1.06196	2.23	1.19651	0.03	
10	1,1,1,1,1,1,7,1,1,1	1.03909	0.03	1.19668	0.01	resonance/epithermal
8	1,1,1,1,1,1,7,3	1.04162	0.27	1.19608	0.06	selected few group collapses
4	5,6,2,3	1.05069	1.15	1.19673	0.01	
2	13,3	1.06608	2.63	1.19593	0.08	

Table 6, Results from collapsing the H-R 16-group cross section set for a U(4.89%)O_2F_2-H_2O, water reflected sphere with H/X=524.

#of groups	cross sections/ICOL	Keff	|%Δ|	K_∞	|%Δ|	remarks
16	H-R 16 group set	1.01521	0.00	1.38523	0.00	
16	16 groups/manual input	1.01535	0.01	1.38542	0.01	
15	2,1,1,1,...1	1.01555	0.03	1.38539	0.01	
14	3,1,1,1,...1	1.01831	0.31	1.38545	0.02	
13	4,1,1,1,...1	1.02255	0.72	1.38531	0.01	fast group collapse [fission spectrum; groups 1-6]
12	5,1,1,1,...1	1.02922	1.38	1.38532	0.01	
11	6,1,1,1,...1	1.03734	2.18	1.38545	0.02	
15	1,1,1,...1,2	1.01918	0.39	1.38497	0.02	
14	1,1,1,...1,3	1.02286	0.75	1.38618	0.07	thermal/epithermal collapse [groups 13-16]
13	1,1,1,...1,4	1.02514	0.98	1.38487	0.03	
15	1,1,1,1,1,1,1,1,1,1,2,1,1,1,1	1.01534	0.01	1.38534	0.01	
14	1,1,1,1,1,1,1,1,1,3,1,1,1,1	1.01554	0.03	1.38532	0.01	
13	1,1,1,1,1,1,1,1,4,1,1,1,1	1.01549	0.03	1.38482	0.03	resonance region collapse [groups 7-12]
12	1,1,1,1,1,1,1,5,1,1,1,1	1.01574	0.05	1.38490	0.02	
11	1,1,1,1,1,1,6,1,1,1,1	1.01594	0.07	1.38483	0.03	
10	1,1,1,1,1,7,1,1,1,1	1.01670	0.15	1.38447	0.05	
9	1,1,1,1,8,1,1,1,1	1.02055	0.53	1.38441	0.06	
8	1,1,1,9,1,1,1,1	1.02653	1.12	1.38434	0.06	resonance region collapse expanded into fast region
7	1,1,10,1,1,1,1	1.03184	1.63	1.38433	0.06	
6	1,11,1,1,1,1	1.05272	3.69	1.38468	0.04	
5	12,1,1,1,1	1.06329	4.74	1.38463	0.04	
10	1,1,1,1,1,1,7,1,1,1	1.01596	0.07	1.38382	0.10	resonance/epithermal
8	1,1,1,1,1,1,7,3	1.02350	0.82	1.38429	0.07	selected few group collapses
2	13,3	1.07540	5.93	1.38433	0.06	

What these two tables confirm is that the majority of the error associated with group collapsing comes from the fast region. At higher neutron energies, the significant interaction is scattering. The higher the neutron energy the more anisotropic or forward preferential the scatting becomes (laboratory system).

To analyze the range of neutron energies where anisotropic scattering is predominate, we will start with Schrodinger's Equation,

$$\nabla^2\Psi(r,\theta,\phi) + \frac{2\mu}{h^2}E\Psi(r,\theta,\phi) = 0\,, \tag{60}$$

where $h \equiv$ the reduced Plank's constant,

$\mu \equiv$ the reduced mass of the system $= \dfrac{Mm}{M+m}$,

$E \equiv$ the center of mass energy of the particles,

to describe the scattering interactions of the neutrons. Using separation of variables and Legendre polynomials, the solution to equation 60, in spherical coordinates is

$$\frac{1}{r^2}\frac{\partial}{\partial r}\left(r^2\frac{\partial R}{\partial r}\right) + \frac{2\mu}{h^2}\left(E - \frac{h^2 l(l+1)}{2\mu r^2}\right)R = 0\,, \tag{61}$$

where R is a function of r,

l is the order of Legendre expansion.

Letting

$$E' = E - \frac{l(l+1)h^2}{2\mu r^2}\,, \tag{62}$$

we see that E' must be greater than or equal to zero to define a real, physical neutron-nucleus interaction. Therefore, we can set E' = 0 to determine the acceptable values for l. Recalling that l = 0 corresponds to isotropic scattering and l = 1 to linear anisotropic scattering, we set l = 1 to determine the energy of neutrons at which scattering shifts from isotropic to linear anisotropic. This relationship is:

$$E = \frac{h^2}{\mu r^2}. \tag{63}$$

We can replace r^2 by the classical radius of the nucleus, R_S, which can be calculated by

$$R_s = 0.15\times10^{-12}A^{1/3}\ \text{cm}. \tag{64}$$

Then the energy in the center of mass system below which a neutron will experience isotropic scattering is:

$$E = \frac{h^2}{\mu\left(0.15\times10^{-12}\right)^2 A^{2/3}}. \tag{65}$$

Roughly we can apply the opposite of this in the laboratory system, the frame of reference in which we are dealing. Therefore, neutrons below the energy given by equation 65 will predominately experience linear anisotropic scattering in the laboratory system. For example, equation 65 gives energies on the order of magnitude of 100 Mev, 6 Mev, and 0.6 Mev for hydrogen, carbon, and uranium respectively. Therefore, neutrons in the energy range of our interest (0 - 10 MeV) are predominately scattered anisotropically.

What this means is that the anisotropic behavior not only depends on neutron energy but on the atomic number of the scatterer as well. The lighter the scatterer and the higher the neutron energy the more anisotropic the scattering interaction (laboratory system).

This is the reason we see the large error in the group collapsed cross sections' K_{eff} results and why most of the error occurs in the top 6 energy groups where the anisotropic affect is most predominate. It also explains why, in Table 3, we see a greater amount of error in problems BRENT1, BRENT2, BRENT3, and BRENT11 (hydrogen moderated systems). Problem BRENT50, a carbon moderated system has less error associated with the collapse than the hydrogen moderated systems but more than BRENT60 a pure uranium metal system.

To further support these observations and conclusions, we see in Tables 5 and 6 that no matter how the cross sections are collapsed the K_{∞}s are virtually the same. In an infinite system, scattering interactions can be treated as truly isotropic. The infinite fluxes are constant over space and representative of an equal number of neutrons moving in every direction. Therefore, if we completely eliminate the angular dependence, as in an infinite medium, the collapse procedure is very accurate.

Since we require the 2-group cross sections, we need to verify that the collapse technique preserves numbers of neutrons and process interaction rates. To do this we compare the scalar flux profiles from calculations with the uncollapsed H-R 16-group cross sections to those with the group collapsed cross sections. For comparison, the flux profiles from the uncollapsed cross sections are summed into the same ICOL as the corresponding group collapsed calculation, thus representing the same total number of neutrons in each group collapsed energy span. Both flux profiles are normalized to 1 for the last (slowest) energy group at the center of the system.

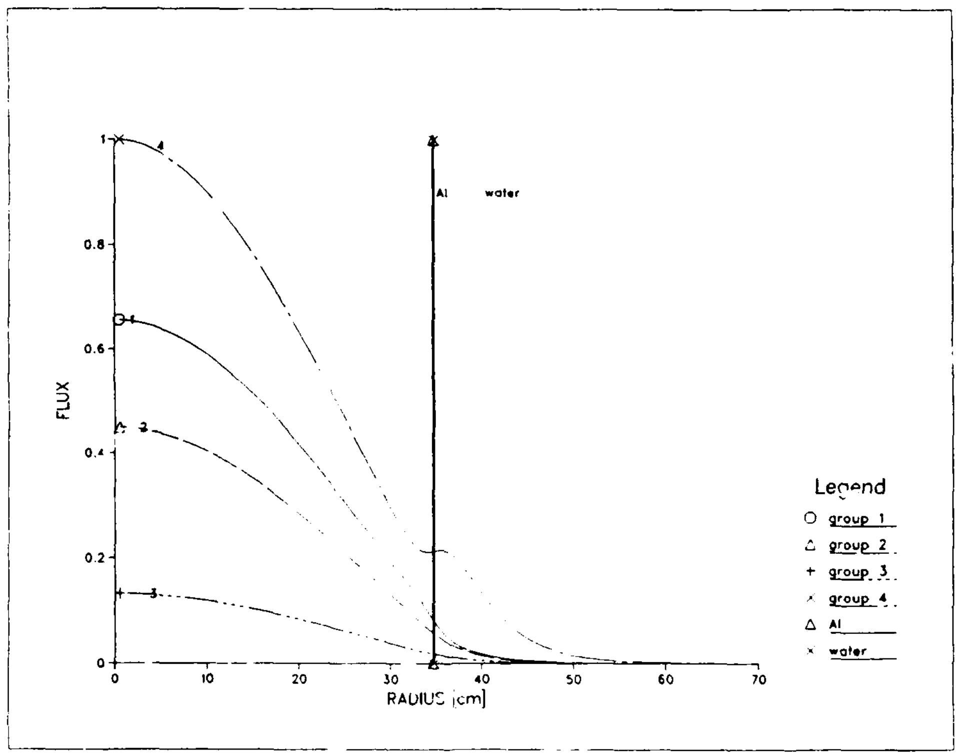

Figure 24, BRENT1, 16 energy groups summed into 4 broad groups; ICOL=5,6,3,2.

to determine a broad group diffusion coefficient does not result in

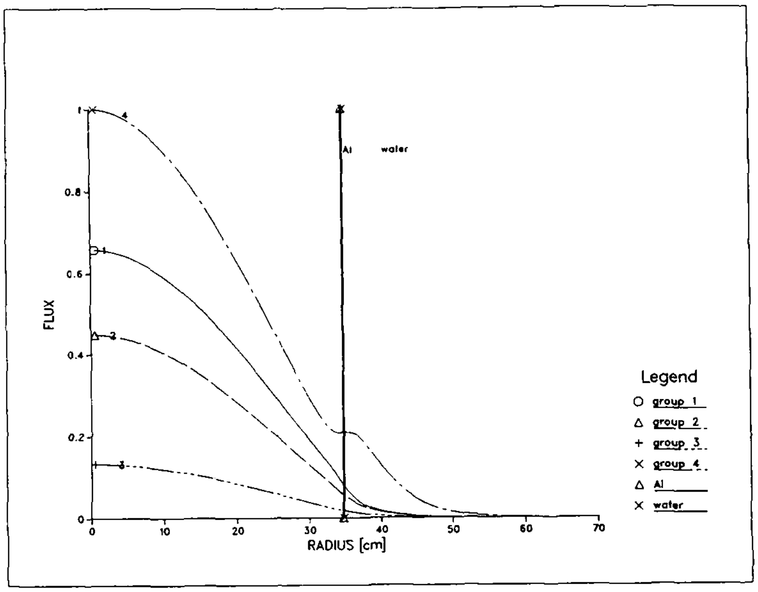

Figure 25, BRENT1 collapsed into 4 energy groups; ICOL=5,6,3,2.

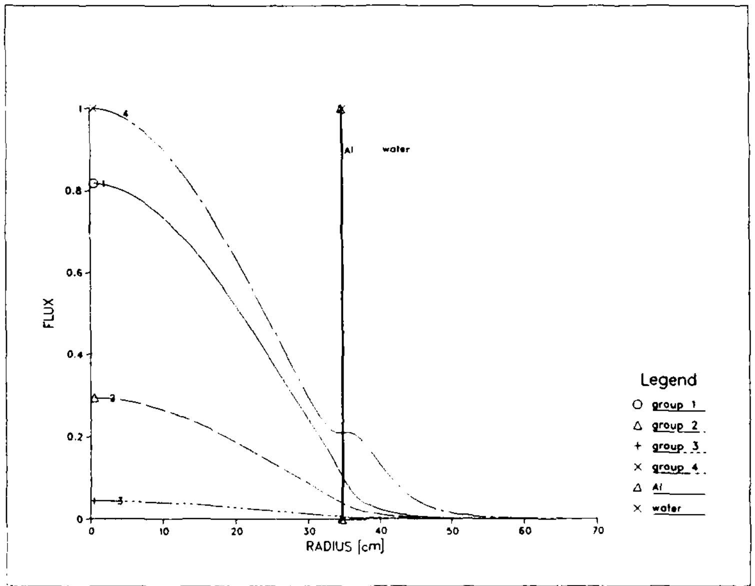

Figure 26, BRENT1, 16 energy groups summed into 4 broad groups; ICOL=7,5,1,3.

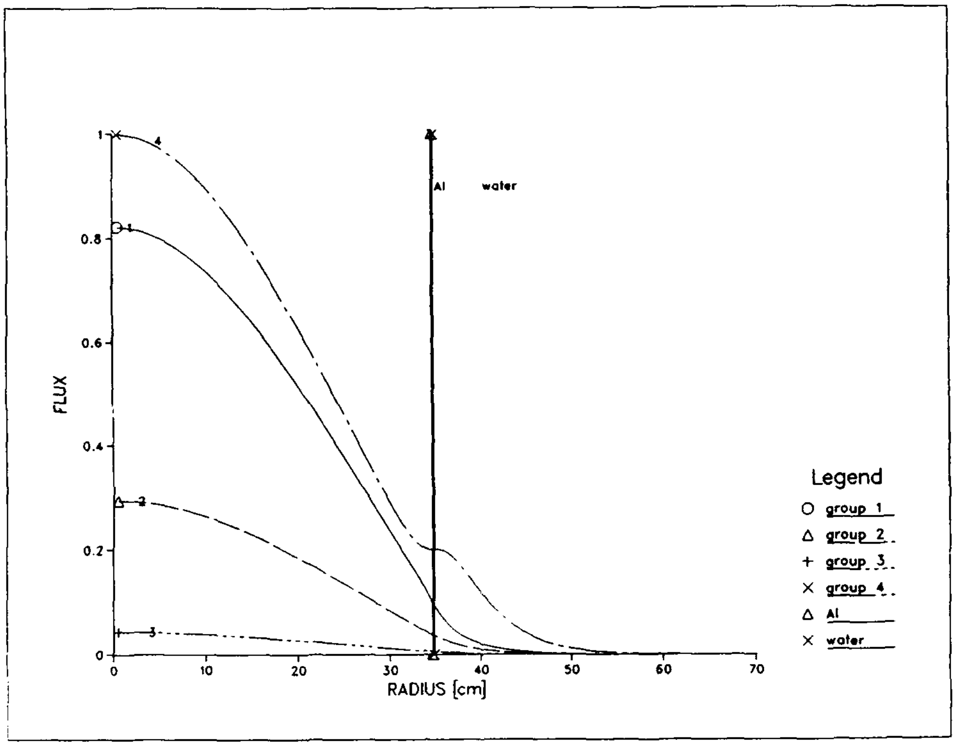

Figure 27, BRENT1 collapsed into 4 energy groups; ICOL=7,5,1,3.

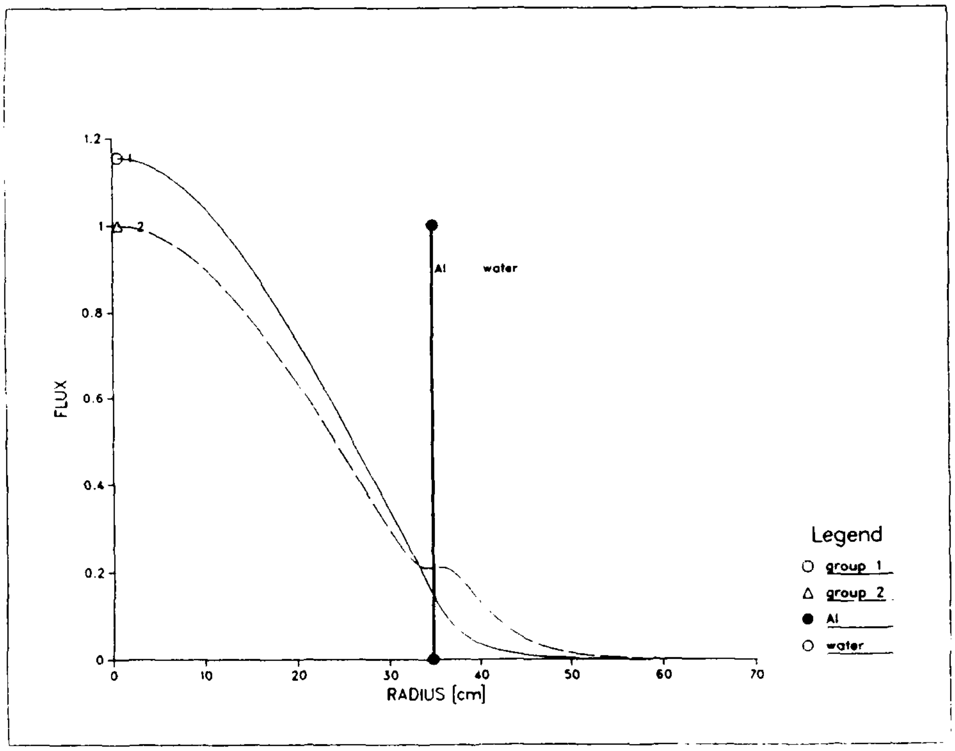

Figure 28, BRENT1, 16 energy groups summed into 2 broad groups; ICOL=13,3.

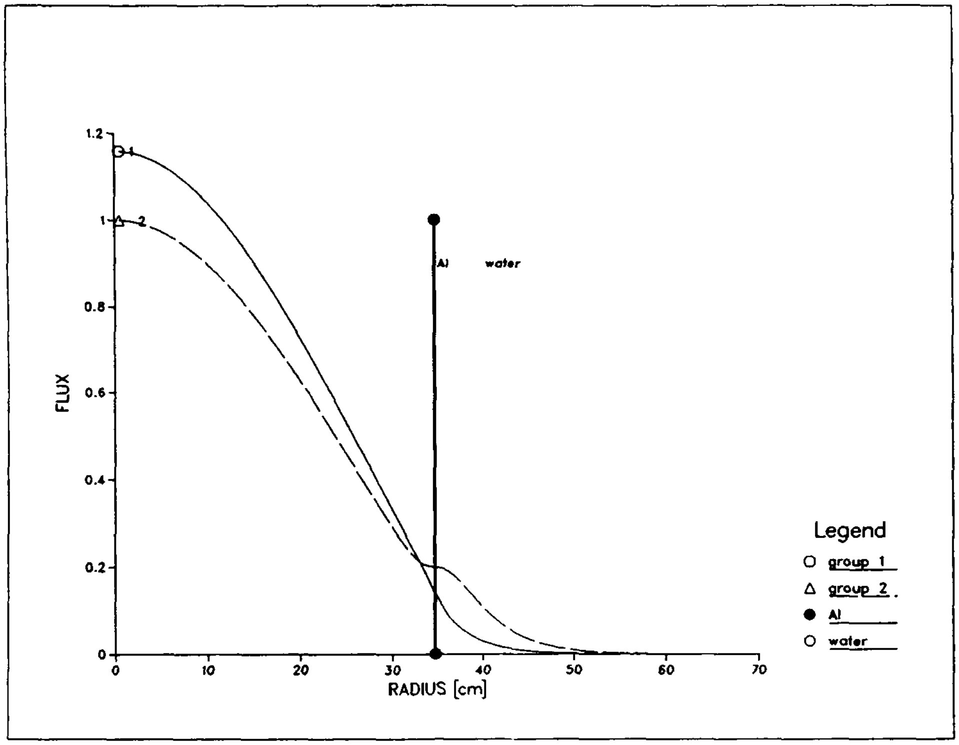

Figure 29, BRENT1 collapsed into 2 enegy groups; ICOL=13,3.

Figures 24 through 29 show comparisons for some different collapses. What is consistent throughout is that the group collapsed cross sections' flux profiles do not vary significantly from the H-R 16-group cross sections' flux profiles. Therefore, the number of neutrons are preserved in the group collapsing process used.

To determine if reaction rates are being conserved in the collapse process, we make use of the System Balance Tables provided in TWODANT's output. These tables tell us, by energy group, how many neutrons are under going each type of interaction. Again we sum the results from the 16-group cross sections into the same ICOL of the corresponding collapse. This and the balance table from the corresponding collapsed calculation are then normalized to a total fission source of 1.

Tables 7 and 8 are a comparison of one system. We see extremely close agreement in all interactions. This show that, with minor error, the overall breakdown of interactions does not change in the collapsing process and for the most part the interaction rates are being preserved. Note, however, that the scattering listed in these tables is the total fraction of neutrons being scattered without regard to their directional dependence (a scalar representation).

Table 7, Balance table from problem BRENT1 TWODANT run with H-R 16 group cross sections summed into 2 groups with ICOL=13,3 and normalized to a fission source of 1.

Energy Group	Fission source	In-Scatter	Self-Scatter	Out-Scatter	Absorption
1-13	1.0	0	13.01489	0.9186828	0.0813172
14-16	0	0.9186828	64.87090	0	0.9186827

Table 8, Balance table from problem BRENT1 TWODANT run with group collapsed ICOL=13,3 2 group cross sections normalized to a fission source of 1.

Energy Group	Fission source	In-Scatter	Self-Scatter	Out-Scatter	Absorption
1	1.0	0	13.10990	0.9167549	0.0832646
2	0	0.9167549	62.33466	0	0.9164495

We have tried to account for the anisotropic scattering behavior by group collapsing the P_1 cross sections in each system. Based on the above results, we see that this has not adequately preserved the directional dependence of the scattering interactions. An additional check is to repeat some of the above calculations with ISCT (the Legendre order of expansion) equal to one. Also, since TWODANT utilizes an S_n, quadrature approach for treatment of angle dependence, we can try to vary the ISN (number of quadrature directions).

Table 9, TWODANT K_{eff} for problem BRENT10 with the H-R 16 group cross section set and 2 group collapsed cross sections (ICOL=13,3) with ISCT=0 and varying ISN.

# of energy groups	ISN	Keff
16	8	1.07856
16	16	1.07829
2	8	1.32810
2	16	1.32809

Table 9 shows the results of this check method. What is obvious is the error in the group collapsed results remain large. We also see there is little difference in the two group determined K_{eff} based on the number of quadrature directions used in determining the fluxes to be used in the collapse process. The collapse process uses the scalar fluxes. The scalar fluxes are nothing more than the sum of the angular or quadrature fluxes. Therefore, direction will not be preserved based on the number of quadrature directions selected because in using scalar fluxes we have neglected most of the angular dependence.

The only way to ensure directional dependence is preserved is to do a space/energy/direction collapse; use the discretized angular fluxes in the collapse process to derive a set of cross sections that are angularly dependent. This would result in several cross sections for a given point in space for a given energy, which is a nasty proposition. Additionally, there is not a method available to insert a cross section set of this nature into TWODANT to verify the results. It is also not practical for us to use a complicated cross section set of this nature in our "simple" two-group diffusion theory approach.

The main conclusion from this work is that the group collapse methods are unable to adequately handle the angular/anisotropic nature of neutron scattering interactions. The group collapsed cross sections cannot be used to do K_{eff} calculations with any reasonable assurance of accuracy. This, however, may not affect the utility of the group collapse method in "thermalness" calculations as will be discussed next.

C. Group Collapsed Cross Sections In Two-Group Diffusion Theory.

Unlike the transport theory method used to determine the fine group fluxes for use in the group collapsing, diffusion theory relaxes the angular dependence. Since the majority of the error associated with the group collapse method is in the treatment of the angular dependence, the collapsed cross sections should be adequate for use in diffusion calculations. Diffusion theory uses the diffusion coefficient, D, to relate the neutron current (directional term) to the gradient of the flux. So the correct calculation and collapsing of D should adequately treat the relaxed angular dependence in the diffusion theory.

Applying the method of calculating and group collapsing the diffusion coefficient presented earlier, the volume averaged flux group collapse technique, and the methods of determining buckling presented in sections 3.1, we calculate K_{eff} for several unreflected systems using the derived two-group analogy to the six-factor formula. The results of this evaluation are presented in Table 10.

Table 10, Comparison of calculated K_{eff}s from TWODANT/H-R 16-group cross sections and diffusion theory/group collapsed 2-group cross sections.

PROBLEM	TWODANT 16-GROUP K_{eff}	DIFFUSION THEORY 2-GROUP K_{eff}	% DIFFERENCE
BRENT10	0.984257	0.986627	+0.24
BRENT11	0.992137	1.002758	+1.07
BRENT12	0.990456	0.997935	+0.76
BRENT13	0.955103	0.931612	-2.46
BRENT14	0.985928	0.966664	-1.95
BRENT15	0.970805	0.953782	-1.75
BRENT16	0.967166	0.946143	-2.17
BRENT20A	0.972672	0.979596	+0.71
BRENT20B	1.011610	1.020979	+0.93
BRENT20C	0.981791	0.990152	+0.23
BRENT20D	0.997210	1.009725	+1.26
BRENT20E	0.990762	1.004361	+1.37
BRENT20F	1.002570	1.014947	+1.23

With the differences in K_{eff} between the 16-group transport and 2-group diffusion calculations ranging from 2.49% to less than 0.10% we are relatively comfortable with our approximations. We are also assured that within the applicability of diffusion theory our calculations of the factors of the six-factor formula for K_{eff} will be within acceptable limits for the simple hand calculation technique we are attempting to provide.

4. RESULTS.

As detailed earlier, we have selected $p/(\eta_2{}^*f_2)$ as the parameter to describe the moderating capability or "thermalness" of a system. The slowing down characteristics of a system are adequately described by p, while η_2 and f_2 account for the absorption characteristics. All of the important characteristics of the slowing down process are present in the thermalness factor, $p/(\eta_2{}^*f_2)$. A plot of this parameter versus critical mass and volume also serves to predict minimum critical mass and volume, an important concept in nuclear criticality safety.

The thermalness factor, $p/(\eta_2{}^*f_2)$, for each of the systems evaluated are presented in graphical format versus critical mass and volume in Figures 30 through 33. As expected $p/(\eta_2{}^*f_2)$ does predict the minimum critical mass and volume for each evaluated fuel-moderator mixture.

The range of $p/(\eta_2{}^*f_2)$ corresponding to minimum critical mass is 0.52 to 0.58. Comparing Figures 30 and 31 we find that this range holds for both bare and reflected sytems. The shapes and lowest points of the curves for the bare and reflected systems are nearly identical. The difference is the magnitude of the critical masses. As expected, the reflected systems have less critical mass than the bare systems. This difference in mass, due to the reflector savings, is the corresponding change of placement of the curves with respect to the critical mass axis. This leads us to conclude that the variation in flux profiles, used in the group collapsing process, between the reflected and bare systems, though producing a slight difference in cross sections for the two situations, does not affect the thermalness factor.

The value of $p/(\eta_2 * f_2)$ corresponding to minimum critical volume ranges from 0.42 to 0.52 for the bare systems and from 0.39 to 0.52 for the reflected systems. Looking at Figures 32 and 33 we see that the difference between the bare and reflected ranges is isolated to the U(30.3%)O_2F_2-H_2O systems. As with the critical mass comparisons, the other three systems have nearly identical curve shapes and minimum points. The problem with the 30.3% enriched systems is an inconsistency found in the literature. J. C. Smith, et al. [Reference 12] reported that the minimum critical cylindrical volume for these systems occurred at an H/X ratio of between 120 and 130 for reflected systems (depending on the radius of the cylinder) and an H/X ratio of 130 for bare cylinders (independent of radius). However, the buckling conversions to spheres reported by Paxton and Pruvost [Reference 1] do not reflect these results. Based on this, the volumes for the two systems with lowest $p/(\eta_2 * f_2)$ factors were recalculated using TWODANT eigenvalue searches. A similar discrepancy is not present for the H/X ratio corresponding to minimum critical mass. The problem is twofold. The first is suspicion of the buckling conversion from the cylinders to spheres. Without knowing the source of extrapolation distances used we cannot confirm or deny either set of results. Second, the converted sphere data is from a combination of 8, 12, and 16 inch diameter cylinders, each system with unique characteristic, such as H/X corresponding to the minimum critical volume. Conversion to spheres will elliminate the uniques, but from the data provided we cannot isolate, only suspect where the true minimum occurs in spherical geometry. The bottom line is suspicion of the $U(30.3\%)O_2F_2$-H_2O data at the lower H/X and $p/(\eta_2{}^*f_2)$ values.

The wider range of $p/(\eta_2{}^*f_2)$ defining minimum critical volume is a factor of enrichment. Enrichment has a larger effect on the critical volume than on the critical mass. Since the $p/(\eta_2{}^*f_2)$ range is larger for minimum critical volume than minimum critical mass, we can assume that we have not adequately treated the enrichment effect on the systems. The range of $p/(\eta_2{}^*f_2)$ which defines minimum critical volume is still narrower or better defined than the the range of H/X for comparable systems. (Our results are compared with Figure 2). Also, within the limits of the six-factor formula, though enrichment does affect the values of η and f slightly, enrichment has a predominant impact on the resonance escape probability. The resonance escape probability is directly related to enrichment. The less U^{238}, or the higher the enrichment, the less resonance absorption and the higher the resonance escape probability. Therefore, in this approach to defining a thermalness parameter, we have described the effect of enrichment in the best manner possible.

One other thing to note is that for a small change in mass of U^{235} there can be a large change in volume. For the lower U^{235} density systems, it requires a substantial change in volume to slightly increase U^{235} mass. The opposite is true at higher densities. Since critical mass and volume are interrelated, for a small change in critical mass, based on the enrichment and fuel to moderator ratio, we will see a larger corresponding change in volume. Therefore, based on a parameter describing moderation and enrichment, we expect to see a wider range of values corresponding to minimum critical volume.

Overall, the results confirm that $p/(\eta_2{}^{*}f_2)$ does an excellent job in defining mass limitations. With a more detailed and accurate critical experimental data base, the critical volume applications look equally promising.

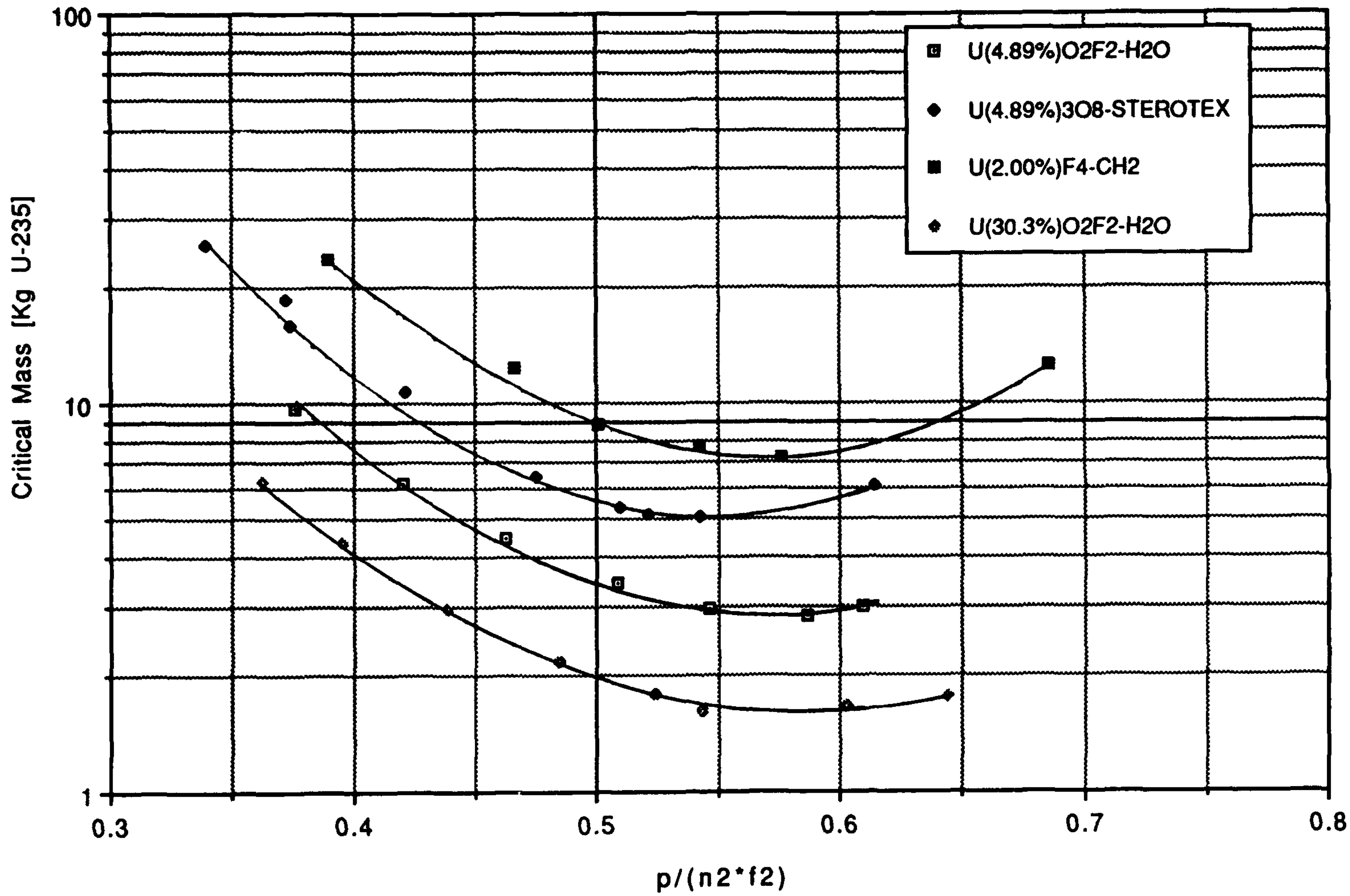

Figure 30, p/(n2*f2) versus critical mass for the bare fuel-moderator mixtures evaluated.

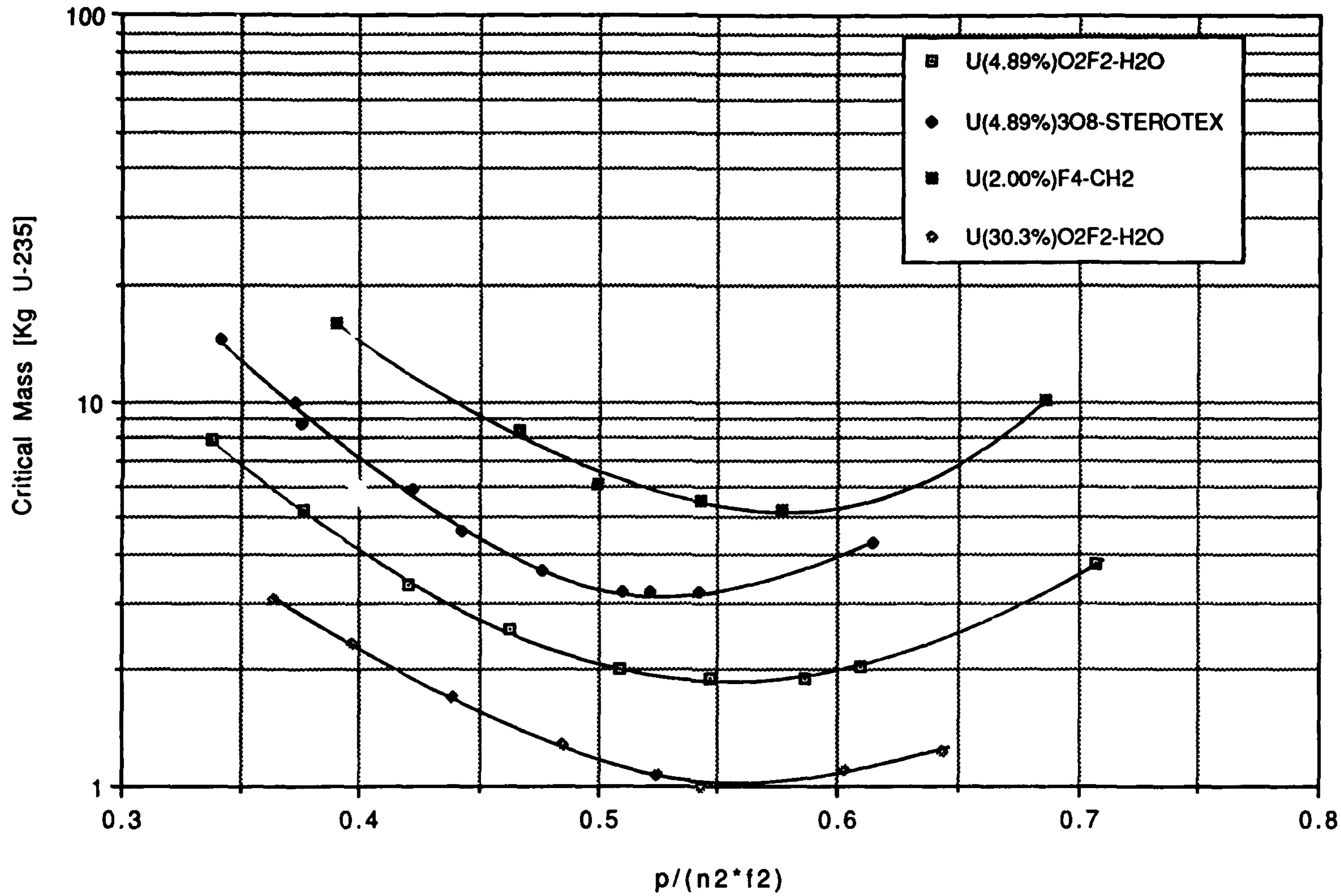

Figure 31, p/(n2*f2) versus critical mass for the reflected fuel-moderator mixtures evaluated.

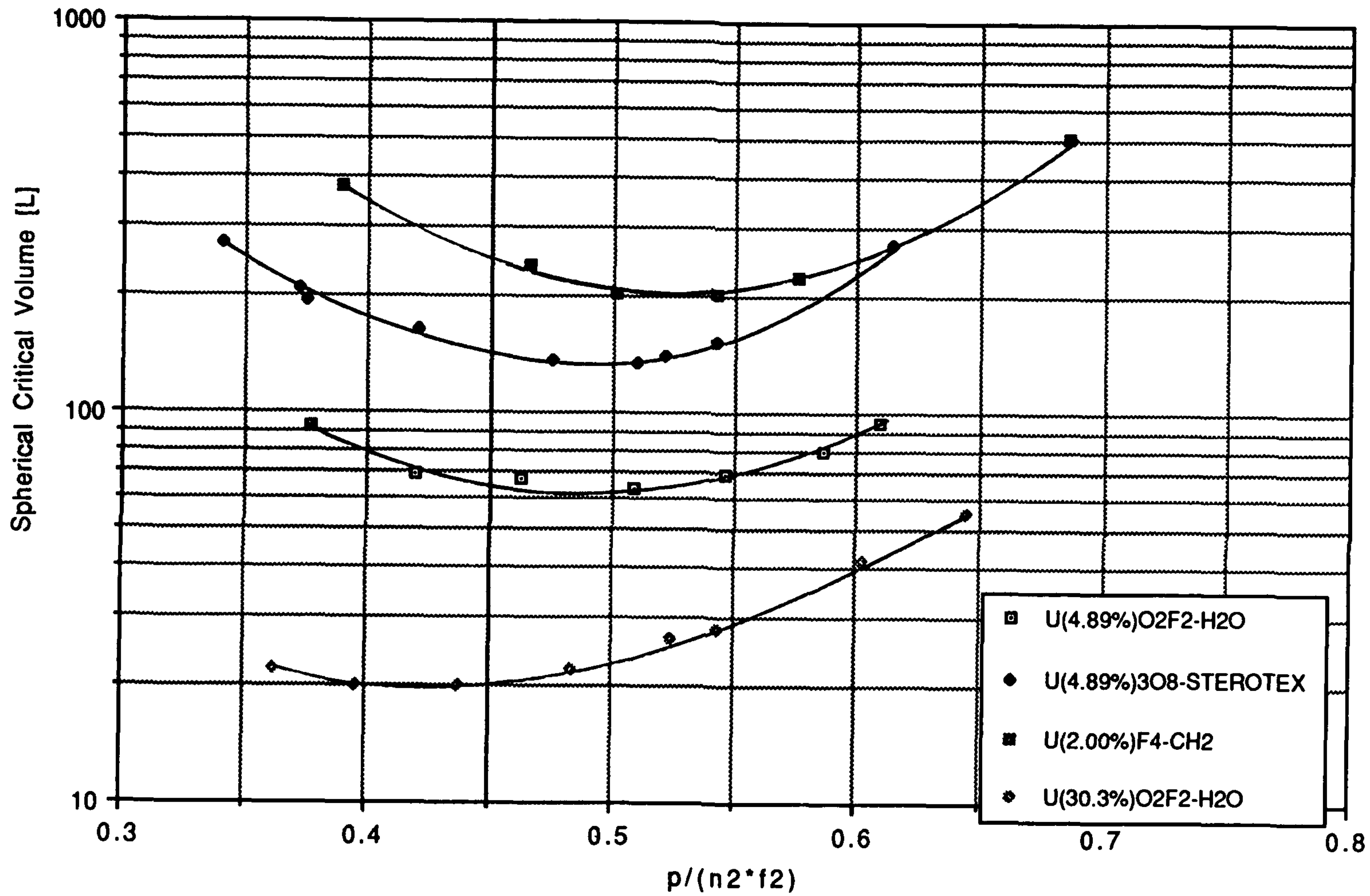

Figure 32, p/(n2*f2) versus spherical critical volume for the bare fuel-moderator mixtures evaluated.

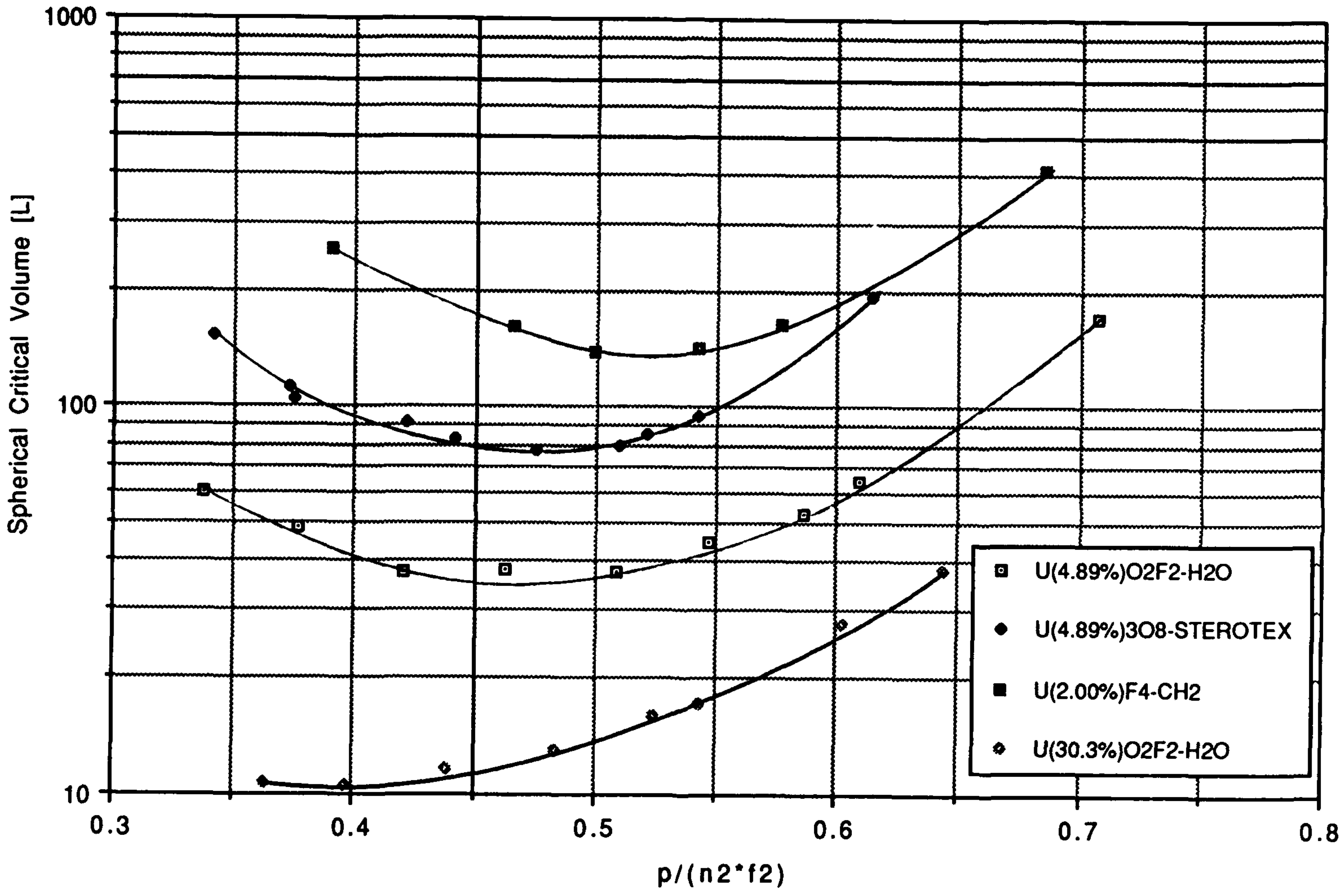

Figure 33, p/(n2*f2) versus spherical critical volume for the reflected fuel-moderator mixtures evaluated.

5. CONCLUSIONS.

Our use of $p/(\eta_2{}^*f_2)$ to define the "thermalness" of a system is as accurate a definition as possible and takes into account the factors important in a moderated system. Based on the limited results of this study, we are comfortable with the potential of $p/(\eta_2{}^*f_2)$ to define mass and volume limits. The same range of $p/(\eta_2{}^*f_2)$ applies to both reflected and bare systems which is very encouraging. The main advantage is its treatment of all of the potential scattering interactions as well as the absorption characteristics important in describing the moderating process rather than just the number of hydrogen atoms to fissile atoms. Before application in any capacity, much more benchmarking and correlating must be done to ensure the precise ranges of $p/(\eta_2{}^*f_2)$ are defined and applicable to the appropriate sytems.

Some very important supporting topics have been brought to light during this study. The first is the problem with the flux weighted group collapse method's ability to maintain directional dependence for reapplication in fewer energy group transport calculations. Though a very appropriate and accurate technique for deriving few group cross sections for diffusion theory calculations, more study needs done to extend this method to account for anisotropic situations.

The final and most important topic is the inadequacy of the critical experiments to be applied to parametric studies and/or validation of numerical codes. There are many examples of important experimental results that cannot be accurately applied because of incomplete data. An example is the $U(30.3\%)O_2F_2$-H_2O data, which does not specify the composition or type of stainless steel used for the fissile solution container. Additionally, we must be careful in blindly using data converted from one geometry to another without knowing how and with what assumptions these conversions were done. There is a valid need for a standard, detailed, and complete set of benchmark criticals for use in the criticality safety business.

We have attempted to provide a useful and better way to characterize thermal systems and avoid an unwanted critical situation. Our hope is that it may assist planners, designers, and technicians in keeping the nuclear industry safe and prosperous.

APPENDIX 1, Tabulation of Evaluated Critical Systems.

Table 11, Homogeneous UO_2F_2-H_2O, Water Reflected Spheres. (4.89 wt% U^{235} enriched uranium. Contained in a 0.1587 cm thick Al spherical shell. 137.16 cm thick reflector)

Reference Name	H/U^{235}	U^{235} Density (g/cm^3)	Volume (L)	Radius (cm)	Mass (Kg U^{235})	TWODANT H-R K_{eff}	2-Group Diffusion K_{eff}	Ref
BRENT1	1099	0.02211	170.5	34.671	3.77	1.03889	1.06610	1,9
BRENT2	524	0.04254	44.9	22.047	1.91	1.01522	1.06110	1,9
BRENT2A[a]	400	0.05347	37.4	20.748	2.00	1.00007	1.04830	9,10
BRENT2B[a]	300	0.06686	38.2	20.900	2.55	1.00008	1.04621	9,10
BRENT2C[a]	200	0.08900	37.5	20.764	3.34	0.99994	1.04654	9,10
BRENT2D[a]	100	0.13257	59.9	24.267	7.94	0.99994	1.04871	9,10
BRENT2E[a]	150	0.10656	49.0	22.693	5.22	0.99994	1.04419	9,10
BRENT3	735	0.03179	64.6	24.943	2.05	1.00017	1.04385	1,9
BRENT4	643	0.03562	53.4	23.302	1.90	0.99779	1.04425	1,9

a. Critical experiments were unavailable at these H/Xs. This data was determined using correlations for UO_2F_2-H_2O solution densities presented by Hugh Clark, Reference 10, and the radius search capability of TWODANT.

Table 12, Homogeneous UO_2F_2-H_2O, Spheres.
(4.89 wt% U^{235} enriched uranium. Contained in a 0.1587 cm thick Al spherical shell)

Reference Name	H/U^{235}	U^{235} Density (g/cm^3)	Volume (L)	Radius (cm)	Mass (Kg U^{235})	TWODANT H-R K_{eff}	2-Group Diffusion K_{eff}	Ref
BRENT10	524	0.04254	69.3	25.444	2.94	0.98426	0.98663	1,9
BRENT10A[a]	400	0.05347	63.9	24.807	3.42	1.00003	0.99834	9,10
BRENT10B[a]	300	0.06686	67.4	25.246	4.51	1.00004	0.99560	9,10
BRENT10C[a]	200	0.08900	69.5	25.501	6.18	0.99993	0.98955	9,10
BRENT10D[a]	150	0.10656	91.3	27.934	9.73	1.00001	0.99177	9,10
BRENT11	735	0.03179	93.9	28.206	2.98	0.99214	1.00276	1,9
BRENT12	643	0.03562	79.9	26.730	2.84	0.99046	0.99794	1,9

a. Critical experiments were unavailable at these H/Xs. This data was determined using correlations for UO_2F_2-H_2O solution densities presented by Hugh Clark, Reference 10, and the radius search capability of TWODANT.

Table 13, Homogeneous U_3O_8-STEROTEX, Water Reflected Spheres.
(4.89 wt% U^{235} enriched uranium. 137.16 cm thick reflector)

Reference Name	H/U^{235}	U^{235} Density (g/cm^3)	Volume (L)	Radius (cm)	Mass (Kg U^{235})	TWODANT H-R K_{eff}	2-Group Diffusion K_{eff}	Ref
BRENT5	102	0.09450	152.5	33.107	14.4	0.98975	1.06021	1,9
BRENT6	124	0.08946	111.9	29.903	10.0	1.00805	1.06819	1,9
BRENT7	147	0.08280	104.7	29.266	8.7	0.98993	1.04269	1,9
BRENT8	199	0.06490	90.9	27.903	5.9	0.99230	1.04532	1,9
BRENT9	245	0.05600	83.4	27.178	4.6	0.97204	1.02338	1,9
BRENT9A	320	0.04800	76.9	26.416	3.6	0.98532	1.03268	1,9
BRENT9B	396	0.04040	79.5	26.674	3.2	0.98964	1.03715	1,9
BRENT9C	449	0.03743	85.0	27.276	3.2	0.99201	1.03748	1,9
BRENT9D	504	0.03340	95.4	28.346	3.2	0.99539	0.98485	1,9
BRENT9E	757	0.02220	194.8	35.961	4.3	1.00858	1.00676	1,9

Table 14, Homogeneous U_3O_8-STEROTEX, Spheres.
(4.89 wt% U^{235} enriched uranium)

Reference Name	H/U^{235}	U^{235} Density (g/cm^3)	Volume (L)	Radius (cm)	Mass (Kg U^{235})	TWODANT H-R K_{eff}	2-Group Diffusion K_{eff}	Ref
BRENT13	102	0.09450	271.2	40.154	25.6	0.95510	0.93161	1,9
BRENT14	124	0.08946	208.0	36.756	18.6	0.98593	0.96666	1,9
BRENT15	147	0.08280	194.0	35.912	16.0	0.97081	0.95378	1,9
BRENT16	199	0.06490	164.1	33.963	10.7	0.96717	0.94614	1,9
BRENT17	320	0.04800	136.0	31.902	6.5	0.96993	0.95719	1,9
BRENT18	396	0.04040	135.0	31.824	5.4	0.97406	0.96427	1,9
BRENT19	449	0.03743	139.9	32.204	5.2	0.97885	0.97466	1,9
BRENT19A	504	0.03340	151.9	33.099	5.1	0.98087	0.97880	1,9
BRENT19B	757	0.02220	272.8	40.233	6.1	0.99355	1.00265	1,9

Table 15, Homogeneous UF_4-CH_2, Water Reflected Spheres.
(2.00 wt% U^{235} enriched uranium)

Reference Name	H/U^{235}	U^{235} Density (g/cm^3)	Volume (L)	Radius (cm)	Mass (Kg U^{235})	TWODANT H-R K_{eff}	2-Group Diffusion K_{eff}	Ref
BRENT25A	195	0.06167	257.0	39.441	15.9	0.97764	0.98243	1,11
BRENT25B	294	0.05193	161.0	33.748	8.4	1.01480	1.02335	1,11
BRENT25C	406	0.04367	139.0	32.135	6.1	0.98554	0.98976	1,11
BRENT25D	496	0.03875	142.0	32.364	5.5	1.00028	1.01172	1,11
BRENT25E	614	0.03214	163.0	33.887	5.2	0.99327	1.00600	1,11
BRENT25F	972	0.02433	413.0	46.198	10.1	1.00328	1.01554	1,11

a. Reference 11 reports a correction factor for variations in sytem densities in relation to the material densities to account for voids in the assemblies. The values for densities listed here are corrected by this factor.

Table 16, Homogeneous UF_4-CH_2 Spheres.
(2.00 wt% U^{235} enriched uranium)

Reference Name	H/U^{235}	U^{235} Density (g/cm^3)	Volume (L)	Radius (cm)	Mass (Kg U^{235})	TWODANT H-R K_{eff}	2-Group Diffusion K_{eff}	Ref
BRENT20A	195	0.06167	379.0	44.894	23.4	0.97267	0.97960	1,11
BRENT20B	294	0.05193	239.0	38.498	12.4	1.01161	1.02098	1,11
BRENT20C	406	0.04367	202.0	36.399	8.8	0.98179	0.99015	1,11
BRENT20D	496	0.03875	201.0	36.362	7.8	0.99721	1.00973	1,11
BRENT20E	614	0.03214	224.0	37.672	7.2	0.99076	1.00436	1,11
BRENT20F	972	0.02433	513.0	49.654	12.5	1.00257	1.01495	1,11

a. Reference 11 reports a correction factor for variations in sytem densities in relation to the material densities to account for voids in the assemblies. The values for densities listed here are corrected by this factor.

Table 17, Homogeneous UO_2F_2-H_2O, Water Reflected Spheres. (30.3 wt% U^{235} enriched uranium. Contained in a 0.163 cm thick stainless steel container. 17.8 cm. thick reflector)

Reference Name	H/U^{235}	U^{235} Density (g/cm^3)	Volume (L)	Radius (cm)	Mass (Kg U^{235})	TWODANT H-R K_{eff}	2-Group Diffusion K_{eff}	Ref
BRENT40[b]	76.7	0.2880	10.7	13.666	3.08	0.99973	0.94947	1,12
BRENT41[c]	106	0.2200	10.6	13.640	2.34	0.99994	0.93591	1,12
BRENT42	167	0.1460	11.6	14.043	1.70	1.00710	0.94229	1,12
BRENT43	257	0.09780	13.0	14.587	1.28	1.01344	0.97202	1,12
BRENT44	378	0.06750	16.1	15.664	1.08	1.00602	0.98187	1,12
BRENT45	439	0.05840	17.1	15.982	1.00	0.99524	0.97665	1,12
BRENT46	657	0.03940	27.8	18.793	1.10	1.00702	1.01000	1,12
BRENT47	815	0.03170	38.1	20.874	1.24	1.00444	1.01529	1,12

a. The two-group diffusion analysis neglected the stainless container. The addition of the stainless steel container added an average reactivity effect of 0.00865 at the lower H/Xs. At higher H/Xs the stainless steal acts as a poison with an associated negative reactivity.

b. According to Reference 12, the minimum volume occurred between H/X=120-130. The volume listed in Reference 1 for H/X=76.7 was 19.5 L. This value is from a buckling conversion from the cylinderical experiment to a sphere. Without the extrapolation distances used in the calculation I cannot verify this number. Therefore, the volume for H/X=76.7 listed here is derived using a radius search with TWODANT.

c. BRENT41 is listed in Reference 1 as volume=11.6 L. This is still larger than the volume derived for BRENT40. For the same reasons BRENT40 was adjusted, the critical volume for BRENT41 was derived.

Table 18, Homogeneous UO_2F_2-H_2O Spheres.
(30.3 wt% U^{235} enriched uranium. Contained in a 0.163 cm thick stainless steel container)

Reference Name	H/U^{235}	U^{235} Density (g/cm^3)	Volume (L)	Radius (cm)	Mass (Kg U^{235})	TWODANT H-R K_{eff}	2-Group Diffusion K_{eff}	Ref
BRENT30[b]	76.7	0.2880	21.9	17.358	6.31	1.00005	0.93938	1,12
BRENT31	106	0.2200	20.0	16.839	4.38	0.98408	0.92330	1,12
BRENT32	167	0.1460	20.0	16.839	2.93	0.98237	0.92861	1,12
BRENT33	257	0.09780	22.1	17.409	2.16	1.00159	0.96082	1,12
BRENT34	378	0.06750	26.3	18.448	1.77	0.99892	0.97218	1,12
BRENT35	439	0.05840	27.7	18.770	1.62	0.99077	0.96759	1,12
BRENT36	657	0.03940	42.1	21.581	1.66	1.00691	1.00351	1,12
BRENT37	815	0.03170	55.5	23.663	1.76	1.00526	1.01001	1,12

a. The two-group diffusion analysis neglected the stainless container. The addition of the stainless steel container added an average reactivity effect of 0.00865 at the lower H/Xs. At the higher H/Xs, the effect is lessened.

b. According to Reference 12, the minimum volume occurred at H/X=130. The volume listed in Reference 1 for H/X=76.7 was 19.5 L. This value is from a buckling conversion from the cylinderical experiment to a sphere. Without the extrapolation distances used in the calculation I cannot verify this number. Therefore, the volume for H/X=76.7 listed here is derived using a radius search with TWODANT.

Table 19, Computationally Derived Critical Spheres.
(Critical size was determined using TWODANT's dimension search capabilities)

Reference Name	Fuel-Moderator	Enrichment (wt%)	U^{235} Density (g/cm^3)	Volume (L)	Radius (cm)	Mass (Kg U^{235})
BRENT50[a]	U-C	93	0.17530	1182.22	65.595	207.24
BRENT60	U metal	93.71	17.5613	2.76	8.697	48.47

a. Fuel-moderator mixture density to begin calculation from Figure 46, Reference 1.

Appendix 2, Two-Group Six-Factor-Formula FORTRAN 77 Program.

```
* This program takes 2DANT output in 16 energy groups,
* collapses the data into two groups [group1>1ev, group2<1ev],
* and computes the individual factors of the six-factor
* formula [as derived in D&H, p. 437]. Assumes energy
* groups choosen so chi1=1, chi2=0. It reads the 16 group data
* from twodant output from the input file 'xsec.inp'. Input in
* this file is ordered;'problem name'(less than 40 characters),
* type geometry(1=sphere/2=inf cylinder/3=finite cylinder),r and/or
* h, number of isotopes other than fuel in the core,'isotope',atom
* density(use TWODANT isotope names) for each other than fuel
* isotope in the core, nusigf,sigt,siga,sigs g-g,sigs(1) g-g*,
* sigs g-g+1,sigs g-g+2,sigs g-g+3,sigs g-g+4,sigs g-g+5,flux
* for each of 16 groups. [* anisotropic self scatter].
* The subroutine "moderator" figures the macroscopic absorption
* cross section for the other than fuel (fissionable materials)
* in the core inorder to compute eta and f seperately. This is
* not possible using just the TWODANT output, which only gives
* zone cross sections.
*******************************************************
********
* SYMBOLS
*
* nusigf=nu*macroscopic fission cross section.
* sigt=total macroscopic cross section.
* siga=macroscopic absorption cross section.
* sigsxy=macroscopic scattering from group x to y.
* sigr=removal cross section.
* r= physical radius of core.
* h= physical height of core.
* ex=extrapolated distance.
* lambatr= transport mean free path.
* d=diffusion coefficient.
* l=diffusion length.
* bsq=buckling^2.
* f1n1=fast utilization factor*fast eta
```

```
* f2n2=thermal utilization factor*thermal eta.
* pnl1=fast non-leakage probability.
* pnl2=thermal non-leakage probability.
* e=fast fission factor.
* a(i,1)=nusigf(i)          a(i,2)=sigt(i)
* a(i,3)=siga(i)            a(i,4)=sigs g-g
* a(i,5)=sigs1 g-g (anisotropic) a(i,6)=sigs g-g+1
* a(i,7)=sigs g-g+2         a(i,8)=sigs g-g+3
* a(i,9)=sigs g-g+4         a(i,10)=sigs g-g+5
* a(i,11)=flux(i)           i=energy group 1 to 16.
*****************************************************
*************
* open data file and designate variables.
*
      implicit double precision(a-z)
      real nusigf1,nusigf2,lambatr,l1,l2,k,a(16,11),n(10)
      real fnusf(16),fsigt(16),fsiga(16),fsgs0(16),fsgs1(16)
      real fsgs2(16),fsgs3(16),fsgs4(16),fsgs5(16),dg(16),fdg(16)
      character*40 name
      character*6 mod(10)
      open(unit=1,file='xsec.inp',status='old')
      open(unit=2,file='facfor',status='new')
*
* enter input conditions from xsec.inp file.
*
* read in problem name.
      read(1,*) name
      write(*,*) name
* read in geom type; 1=sphere, 2=inf cylinder, 3=finite cylinder.
      read(1,*) geom
* read in core dimensions.
      if(geom.eq.3)then
       read(1,*)r,h
      else
       read(1,*)r
      endif
* read in number of isotopes other than fuel.
      read(1,*)m
* read in the isotopes and atom densities.
      do 5 i=1,m
```

```
      read(1,*)mod(i),n(i)
      write(*,*)mod(i),n(i)
  5  continue
* read in the 16 group cross section and flux array.
    do 15 i=1,16
      read(1,*) (a(i,j),j=1,11)
      write(*,*) a(i,11)
 15  continue
* input data for a reflected system. If reflected it requires
* the reflectors data to be inputed by the input file "refl.inp".
    write(*,*)'Is system reflected? 1=no/2=yes'
    read(*,*) ref
    if(ref.eq.2)then
      write(*,*)'Is reflector water? 1=no/2=yes'
      read(*,*)refm
    endif
*******************************************************
********************
* collapse 16 group data from unit 1 into two groups.
*
* determine sum of the fluxes for the collapsed group 1.
    sflux1=0
    do 110 i=1,13
      sflux1=sflux1+a(i,11)
 110 continue
    write(*,*)'sum fluxes group 1 =',sflux1
* determine sum of the fluxes for the collapsed group 2.
    sflux2=0
    do 120 i=14,16
      sflux2=sflux2+a(i,11)
 120 continue
* multiply all group values by the group flux.
    do 125 i=1,16
      fnusf(i)=a(i,1)*a(i,11)
      fsigt(i)=a(i,2)*a(i,11)
      fsiga(i)=a(i,3)*a(i,11)
      fsgs0(i)=a(i,4)*a(i,11)
      fsgs1(i)=a(i,6)*a(i,11)
      fsgs2(i)=a(i,7)*a(i,11)
      fsgs3(i)=a(i,8)*a(i,11)
```

```
      fsgs4(i)=a(i,9)*a(i,11)
      fsgs5(i)=a(i,10)*a(i,11)
* calculate diffusion coefficient for each group and multiply
* by group flux.
      dg(i)=1.0d00/(3.0d00*(a(i,2)-a(i,5)))
      fdg(i)=dg(i)*a(i,11)
 125 continue
* determine numerators for group 1 calculations.
    snusf1=0
    ssigt1=0
    ssiga1=0
    ssgs01=0
    sd1=0
    do 130 i=1,13
      snusf1=snusf1+fnusf(i)
      ssigt1=ssigt1+fsigt(i)
      ssiga1=ssiga1+fsiga(i)
      ssgs01=ssgs01+fsgs0(i)
      sd1=sd1+fdg(i)
 130 continue
    write(*,*)'snusf1,t1,a1'
    write(*,*)snusf1,ssigt1,ssiga1
* determine numerators for group1 self scatter, sigs11.
    ssgs11=0
    do 140 i=1,12
      ssgs11=ssgs11+fsgs1(i)
 140 continue
    ssgs21=0
    do 150 i=1,11
      ssgs21=ssgs21+fsgs2(i)
 150 continue
    ssgs31=0
    do 160 i=1,10
      ssgs31=ssgs31+fsgs3(i)
 160 continue
    ssgs41=0
    do 170 i=1,9
      ssgs41=ssgs41+fsgs4(i)
 170 continue
    ssgs51=0
```

```
      do 180 i=1,8
        ssgs51=ssgs51+fsgs5(i)
  180 continue
* calculate group 1 constants.
      nusigf1=snusf1/sflux1
      sigt1=ssigt1/sflux1
      siga1=ssiga1/sflux1
      sigs11=(ssgs01+ssgs11+ssgs21+ssgs31+ssgs41+ssgs51)/sflux1
      d1=sd1/sflux1
* calculate sigs12.
      ssgs22=fsgs2(12)+fsgs2(13)
      ssgs32=fsgs3(11)+fsgs3(12)+fsgs3(13)
      ssgs42=fsgs4(10)+fsgs4(11)+fsgs4(12)
      ssgs52=fsgs5(9)+fsgs5(10)+fsgs5(11)
      sigs12=(fsgs1(13)+ssgs22+ssgs32+ssgs42+ssgs52)/sflux1
* determine numerators for group 2 calculations.
      snusf2=0
      ssigt2=0
      ssiga2=0
      ssgs03=0
      sd2=0
      do 190 i=14,16
        snusf2=snusf2+fnusf(i)
        ssigt2=ssigt2+fsigt(i)
        ssiga2=ssiga2+fsiga(i)
        ssgs03=ssgs03+fsgs0(i)
        sd2=sd2+fdg(i)
  190 continue
      ssgs13=fsgs1(14)+fsgs1(15)
* calculate group 2 constants.
      nusigf2=snusf2/sflux2
      sigt2=ssigt2/sflux2
      siga2=ssiga2/sflux2
      sigs22=(ssgs03+ssgs13+fsgs2(14))/sflux2
      d2=sd2/sflux2
*****************************************************
********************
* Calculate removal cross section.
*
      sigr1=siga1+sigs12
```

```
*****************************************************
*********************
* Calculate L^2
*
    l1=d1/sigr1
    l2=d2/siga2
*****************************************************
*********************
* If system is reflected, subroutine reflector calculates the
* buckling of the system.
*
    if(ref.eq.2)then
      call reflector(ssiga1,ssiga2,snusf1,snusf2,sd1,sd2,
   +          sflux1,sflux2,r,bsq,delta,l2,d1,sigs12,refm)
      if(refm.eq.1)then
        go to 20
      elseif(refm.eq.2)then
        r=r+delta
      endif
    endif
*****************************************************
*********************
* calculate buckling^2.  Buckling is determined for one group only
* The two group Ds are collapsed into one group [for un-refl
systems].
*
    pi=3.141592654d00
    r1=r
    h1=h
 10 if(geom.eq.1)then
      bsq=(pi/r1)**2.0d00
    elseif(geom.eq.2)then
      bsq=(2.405d00/r1)**2.0d00
    elseif(geom.eq.3)then
      bsq=((2.405d00/r1)**2.d00)+((pi/h1)**2.d00)
    endif
    if(refm.eq.2)go to 20
*
* calculate extrapolated distance.
*
```

```
* dd is the diffusion coefficient determined from input data.
* d is the one group diffusion coefficient as collapsing
* from d1 and d2.
      dd=(sd1+sd2)/(sflux1+sflux2)
      d=(((d2*bsq+siga2)*d1)+(sigs12*d2))/(d2*bsq+siga2+sigs12)
      lambatr=3.0d00*d
      ex=0.71d00/lambatr
      write(*,*) ex
      if(geom.eq.3)then
        r2=r+ex
        h2=h+2.d00*ex
      else
        r2=r+ex
      endif
*
* check to see if correct extrapolation distances are used.
*
      z=abs(r1-r2)
      if(z.lt.1.0d-3)then
        go to 20
      else
        r1=r2
        h1=h2
        go to 10
      endif
*******************************************************
***********************
* calculate fn, utilization factor*eta.
*
   20 f1n1=nusigf1/sigr1
      f2n2=nusigf2/siga2
*******************************************************
***********************
* calculate p, resonance escape probability.
*
      p=sigs12/sigr1
*******************************************************
***********************
* calculate non-leakage probabilities.
*
```

```
      pnl1=1.0d00/(1.0d00+l1*bsq)
      pnl2=1.0d00/(1.0d00+l2*bsq)
*****************************************************
*********************
* calculate e, fast fission factor.
*
      e=1.0d00+((nusigf1*(siga2+d2*bsq))/(nusigf2*sigs12))
*****************************************************
**********************
* determine siga of the other than fuel material inorder to compute
* eta and f seperately.
*
      call moderator(mod,n,a,sigao1,sigao2,sflux1,sflux2,m)
*
* determine eta and f for the two groups
*
      sigaf1=siga1-sigao1
      sigaf2=siga2-sigao2
      eta1=nusigf1/sigaf1
      f1=sigaf1/sigr1
      eta2=nusigf2/sigaf2
      f2=sigaf2/siga2
*****************************************************
***********************
* verify calculations by determining k using derived factors.
*
      k=f2n2*p*e*pnl1*pnl2
*****************************************************
**********************
* output to data file.
*
      write(2,*) '    ',name
      write(2,*) ''
      write(2,*) '    f1n1=',f1n1,'    f2n2=',f2n2
      write(2,*) ''
      write(2,*) '    eta1=',eta1,'    f1=',f1
      write(2,*) ''
      write(2,*) '    eta2=',eta2,'    f2=',f2
      write(2,*) ''
      write(2,*) '    p=',p,'    fast fission factor=',e
```

```
      write(2,*) ''
      write(2,*) '   PNL1=',pnl1,'    PNL2=',pnl2
      write(2,*) ''
      write(2,*) '   keff=',k
      write(2,*) '   D1=',d1,'D2=',d2,'dd=',dd,'d=',d
      write(2,*) '   nusigf2,sigt2,siga2'
      write(2,*) '   ',nusigf2,sigt2,siga2
      write(2,*) '   nusigf1,sigt1,siga1'
      write(2,*) '   ',nusigf1,sigt1,siga1
      write(2,*) '   sigs11,sigs12,sigs22'
      write(2,*) '   ',sigs11,sigs12,sigs22
      write(2,*) '   sigao2=',sigao2,'sigao1=',sigao1
*****************************************************
**********************
      stop
      end
*****************************************************
**********************
*
*
      subroutine moderator(mod,n,a,sigao1,sigao2,sflux1,sflux2,m)
*
* This subroutine takes the atom densities of each isotope,other
than
* the fuel, in the core and calculates the macroscopic absorption
* cross section for the other than fuel materials in the core.
*
      implicit double precision(a-z)
      real n(10),a(16,11),c(16),siga1(10),siga2(10),fsigai(16)
      character*6 mod(10)
*
* reinitialize sum variables.
      sigao1=0
      sigao2=0
*
* check the isotope and match it to its appropriate cross section
set.
* The following c(1-16) arrays are the sixteen group microscopic
* absorption cross sections for each listed isotope.
*
```

```
      do 10 i=1,m
       if(mod(i).eq.'H')then
*         hydrogen 'H'
         c(1)=0.0
         c(2)=0.0
         c(3)=0.0
         c(4)=0.0
         c(5)=0.0
         c(6)=0.0
         c(7)=0.0
         c(8)=1.000225d-3
         c(9)=4.000187d-3
         c(10)=8.000195d-3
         c(11)=1.400017d-2
         c(12)=2.500021d-2
         c(13)=4.500001d-2
         c(14)=6.999993d-2
         c(15)=1.299999d-1
         c(16)=3.300000d-1
       elseif(mod(i).eq.'O16')then
*         oxygen 'O16'
         c(1)=4.000000d-2
         c(2)=0.0
         c(3)=0.0
         c(4)=0.0
         c(5)=0.0
         c(6)=0.0
         c(7)=0.0
         c(8)=0.0
         c(9)=0.0
         c(10)=0.0
         c(11)=0.0
         c(12)=0.0
         c(13)=0.0
         c(14)=0.0
         c(15)=0.0
         c(16)=2.000033d-4
       elseif(mod(i).eq.'F19')then
*         'F19'
         c(1)=1.000000d-1
```

```
      c(2)=0.0
      c(3)=0.0
      c(4)=2.000332d-4
      c(5)=2.000332d-4
      c(6)=0.0
      c(7)=0.0
      c(8)=0.0
      c(9)=0.0
      c(10)=0.0
      c(11)=0.0
      c(12)=0.0
      c(13)=1.000016d-3
      c(14)=2.000004d-3
      c(15)=4.000008d-3
      c(16)=8.000016d-3
    elseif(mod(i).eq.'C')then
*      carbon 'C'
      c(1)=0.0
      c(2)=0.0
      c(3)=0.0
      c(4)=0.0
      c(5)=0.0
      c(6)=0.0
      c(7)=0.0
      c(8)=0.0
      c(9)=0.0
      c(10)=0.0
      c(11)=0.0
      c(12)=0.0
      c(13)=0.0
      c(14)=0.0
      c(15)=6.999970d-4
      c(16)=2.999961d-3
    elseif(mod(i).eq.'N')then
*      nitrogen 'N'
      c(1)=2.500000d-1
      c(2)=1.100000d-1
      c(3)=4.000002d-2
      c(4)=4.000000d-2
      c(5)=2.000033d-3
```

```
        c(6)=2.000033d-3
        c(7)=4.000067d-3
        c(8)=7.999957d-3
        c(9)=1.899993d-2
        c(10)=4.000008d-2
        c(11)=7.000005d-2
        c(12)=1.200000d-1
        c(13)=2.200000d-1
        c(14)=3.600001d-1
        c(15)=6.400001d-1
        c(16)=1.670000d00
      endif
*
* Compute the two group absorbtion macroscopic cross section for
* each isotope and sum them to get a total other than fuel
* macroscopic cross section.
*
       do 20 j=1,16
        fsigai(j)=c(j)*n(i)*a(j,11)
 20    continue
       sfsai1=0
       do 30 j=1,13
        sfsai1=sfsai1+fsigai(j)
 30    continue
       sfsai2=fsigai(14)+fsigai(15)+fsigai(16)
       siga1(i)=sfsai1/sflux1
       siga2(i)=sfsai2/sflux2
       sigao1=sigao1+siga1(i)
       sigao2=sigao2+siga2(i)
 10 continue
      return
      end
****************************************************
**************
*
*
      subroutine reflector(ssiga1,ssiga2,snusf1,snusf2,
     +      sd1,sd2,sflux1,sflux2,r,bsq,delta,l2,d1,
     +      sigs12,refm)
*
```

```
* this subroutine calculates the reflector saving for a system.
* it reads the reflector data from an input file "REFL.INP".
* The order of data in this input file is; reflector thickness,
* sigt,siga,sigs1 g-g (anisotropic self scatter), and flux.
*
      implicit double precision(a-z)
      real d(16,4),gdr(16),fsigar(16),lr,nusigf,msq,lhs,l2
      character*40 name
      open(unit=3,file='refl.inp',status='old')
*
* Collapse siga,nusigf,and D of the core into one group.
* This is for the calculation of material buckling, used
* as a first guess in solving for the actual buckling.
*
      siga=(ssiga1+ssiga2)/(sflux1+sflux2)
      nusigf=(snusf1+snusf2)/(sflux1+sflux2)
      dd=(sd1+sd2)/(sflux1+sflux2)
*
* Calculate material buckling of the core.
*
      bmc=dsqrt((nusigf-siga)/dd)
      write(*,*) 'bmc=',bmc
*
* Read in and collapse reflector data into one group.
*
      read(3,*) name
      write(*,*)'reflector file is for',name
      read(3,*) b
      do 10 i=1,16
        read(3,*) (d(i,j),j=1,4)
   10 continue
*
* Sum the reflector fluxes.
*
      sfluxr=0.0
      do 20 i=1,16
        sfluxr=sfluxr+d(i,4)
   20 continue
*
* Calculate the diffusion coefficient for the reflector.
```

```
*
      sdr=0.0
      do 30 i=1,16
        gdr(i)=d(i,4)/(3.0d00*(d(i,1)-d(i,3)))
        sdr=sdr+gdr(i)
   30 continue
      dr=sdr/sfluxr
*
* Collapse siga of the reflector into one group.
*
      sfsgar=0.0
      do 40 i=1,16
        fsigar(i)=d(i,2)*d(i,4)
        sfsgar=sfsgar+fsigar(i)
   40 continue
      sigar=sfsgar/sfluxr
*
* Calculate L of the reflector.
*
      lr=dsqrt(dr/sigar)
*
* Calculate the reflector savings.
*
* For a slab reactor this is the equation for reflector savings.
*      delta=(datan(((dd*bmc*lr)/dr)*dtanh(b/lr)))/bmc
* For a reflected sphere(assumes infinite reflection).
*
      if(refm.eq.1)then
       rhs=1.0d00-((dr/dd)*(r/lr+1))
       bmcr=bmc*r
       x=bmcr/10.0d00
   50   do 60 br=bmcr,0.0,-x
        lhs=br/dtan(br)
        if(lhs.gt.0.0)go to 60
        error=abs((lhs-rhs)/rhs)
        if(error.lt.1.0d-4)go to 70
        if(lhs.gt.rhs)then
          bmcr=br+x
          x=x/10.0d00
          go to 50
```

```
        endif
 60     continue
       write(*,*)'reflected buckling didnot converge!!!!!'
 70     buckl=br/r
       bsq=buckl**2.0d00
      elseif(refm.eq.2)then
       tau=d1/sigs12
       msq=l2+tau
       delta=7.2+(0.10*(msq-40.0))
      endif
*
      return
      end
```

Appendix 3, Group Collapsing FORTRAN 77 Program.

```
* This program reads the xsec.inp file as specified in the
* THESIS1.for program (Appendix 2).  It collapses cross sections into
* a user defined
ICOL IAW the theory outlined in Section 3.3.
*
    implicit double precision(a-h)
    implicit double precision(j-m)
    implicit double precision(o-z)
    real nusigf(16)
    dimension a(16,11),ad(10),ic(16),icol(16)
    dimension sflux(16),fnusf(16),fsigt(16),fsiga(16),fsgs0(16)
    dimension fsgs1(16),fsgs2(16),fsgs3(16),fsgs4(16),fsgs5(16)
    dimension sfnusf(16),sfsigt(16),sfsiga(16),sfsgs0(16)
    dimension sfsgs1(16),sfsgs2(16),sfsgs3(16),sfsgs4(16),
               sfsgs5(16)
    dimension sfsgg(16),sfsgg1(16),sfsgg2(16),sfsgg3(16),
               sfsgg4(16)
    dimension sfsgs11(16),sfsgs21(16),sfsgs31(16),sfsgs41(16)
    dimension sfsgs51(16),sfsgs22(16),sfsgs32(16),sfsgs42(16)
    dimension sfsgs52(16),sfsgs33(16),sfsgs43(16),sfsgs53(16)
    dimension sfsgs44(16),sfsgs54(16),sfsgs55(16)
    dimension sigt(16),siga(16),sgsgg(16),sgsgg1(16),sgsgg2(16)
    dimension sgsgg3(16),sgsgg4(16),sgsgg5(16)
    character*40 name
    character*6 mod(10)
    open(unit=1,file='xsec.inp',status='old')
    open(unit=11,file='collapse.dat',status='new')
*****************************************************
******
*
* Read in data from xsec.inp file [generic file for use with thesis1].
*
* problem name
    read(1,*) name
    write(*,*) name
* geom type; 1=sphere, 2=inf cyl, 3= finite cyl
    read(1,*) geom
* core dimensions.
```

```
      if(geom.eq.3)then
        read(1,*) r,h
      else
        read(1,*) r
      endif
* number of isotopes other than fuel.
      read(1,*) m
* isotopes and atom densities of other than fuel material.
      do 5 i=1,m
        read(1,*) mod(i),ad(i)
    5 continue
* 16 group cross sections and flux array as defined above.
      do 10 i=1,16
        read(1,*) (a(i,j),j=1,11)
   10 continue
* from screen input number of broad groups and group breakdown.
   12 write(*,*) 'input number of desired broad groups [NBG]'
      read(*,*) nbg
      ico=0
      do 15 n=1,nbg
        write(*,390) n
        read(*,*) ic(n)
        ico=ico+ic(n)
        icol(n)=ico
        write(*,*) icol(n)
   15 continue
      if(icol(nbg).ne.16)then
        write(*,*)'SUM OF ICOL VALUES .NE. 16'
        goto 12
      endif
* If collapsing microscopic cross section and you want macroscopic
* input the atom density of the applicable isotope. For my purpose
* this is primarily for collapsing of P1 cross sections.
      write(*,*)'IF A MICRO CROSS SECTION SET AND MACRO DESIRED....'
      write(*,*)'INPUT ATOM DENSITY OF ISOTOPE. OTHERWISE INPUT
0.'
      read(*,*)adh
*******************************************************
*************************
* Determine the sum of the fluxes for each broad group
```

```
*
      do 20 n=1,nbg
        sflux(n)=0.0
        if(n-1.eq.0)then
          do 30 i=1,icol(n)
            sflux(n)=sflux(n)+a(i,11)
 30         continue
        else
          do 40 i=icol(n-1)+1,icol(n)
            sflux(n)=sflux(n)+a(i,11)
 40         continue
        endif
 20   continue
**********************************************************
*************************
* Multiply each group value by its corresponding flux.
*
      do 50 i=1,16
        fnusf(i)=a(i,1)*a(i,11)
        fsigt(i)=a(i,2)*a(i,11)
        fsiga(i)=a(i,3)*a(i,11)
        fsgs0(i)=a(i,4)*a(i,11)
        fsgs1(i)=a(i,6)*a(i,11)
        fsgs2(i)=a(i,7)*a(i,11)
        fsgs3(i)=a(i,8)*a(i,11)
        fsgs4(i)=a(i,9)*a(i,11)
        fsgs5(i)=a(i,10)*a(i,11)
 50   continue
**********************************************************
*****************************
* Determine numerators for nusigf,sigt,siga collapse.
* The broad group self scatter contribution from the 16 group
* self scatter will also be summed during this process.
*
      do 60 n=1,nbg
        sfnusf(n)=0.0
        sfsigt(n)=0.0
        sfsiga(n)=0.0
        sfsgs0(n)=0.0
        if(n-1.eq.0)then
```

```
        do 70 i=1,icol(n)
          sfnusf(n)=sfnusf(n)+fnusf(i)
          sfsigt(n)=sfsigt(n)+fsigt(i)
          sfsiga(n)=sfsiga(n)+fsiga(i)
          sfsgs0(n)=sfsgs0(n)+fsgs0(i)
 70       continue
      else
        do 80 i=icol(n-1)+1,icol(n)
          sfnusf(n)=sfnusf(n)+fnusf(i)
          sfsigt(n)=sfsigt(n)+fsigt(i)
          sfsiga(n)=sfsiga(n)+fsiga(i)
          sfsgs0(n)=sfsgs0(n)+fsgs0(i)
 80       continue
      endif
 60   continue
*****************************************************
**************************
* Calculate self scatter numerators for each broad group.
*
    do 90 n=1,nbg
      sfsgg(n)=0.0
      sfsgs1(n)=0.0
      sfsgs2(n)=0.0
      sfsgs3(n)=0.0
      sfsgs4(n)=0.0
      sfsgs5(n)=0.0
      if(n.eq.1)then
        if(ic(n).eq.1)then
          sfsgg(n)=sfsgs0(n)
          goto 90
        elseif(ic(n).eq.2)then
          sfsgg(n)=sfsgs0(n)+fsgs1(1)
          goto 90
        elseif(ic(n).eq.3)then
          sfsgg(n)=sfsgs0(n)+fsgs1(1)+fsgs1(2)+fsgs2(1)
          goto 90
        elseif(ic(n).eq.4)then
          sfsgg(n)=sfsgs0(n)+fsgs1(1)+fsgs1(2)+fsgs1(3)+
     +            fsgs2(1)+fsgs2(2)+fsgs3(1)
          goto 90
```

```
      elseif(ic(n).eq.5)then
       sfsgg(n)=sfsgs0(n)+fsgs1(1)+fsgs1(2)+fsgs1(3)+fsgs1(4)+
     +          fsgs2(1)+fsgs2(2)+fsgs2(3)+fsgs3(1)+fsgs3(2)+
     +          fsgs4(1)
       goto 90
      elseif(ic(n).ge.6)then
       do 100 i=1,icol(n)-1
        sfsgs1(n)=sfsgs1(n)+fsgs1(i)
100     continue
       do 110 i=1,icol(n)-2
        sfsgs2(n)=sfsgs2(n)+fsgs2(i)
110     continue
       do 120 i=1,icol(n)-3
        sfsgs3(n)=sfsgs3(n)+fsgs3(i)
120     continue
       do 130 i=1,icol(n)-4
        sfsgs4(n)=sfsgs4(n)+fsgs4(i)
130     continue
       do 140 i=1,icol(n)-5
        sfsgs5(n)=sfsgs5(n)+fsgs5(i)
140     continue
       sfsgg(n)=sfsgs0(n)+sfsgs1(n)+sfsgs2(n)+sfsgs3(n)+
     +          sfsgs4(n)+sfsgs5(n)
       goto 90
      endif
     elseif(n.ge.2)then
      if(ic(n).le.1)goto 200
      do 150 i=icol(n-1)+1,icol(n)-1
       sfsgs1(n)=sfsgs1(n)+fsgs1(i)
150    continue
      if(ic(n).le.2)goto 200
      do 160 i=icol(n-1)+1,icol(n)-2
       sfsgs2(n)=sfsgs2(N)+fsgs2(i)
160    continue
      if(ic(n).le.3)goto 200
      do 170 i=icol(n-1)+1,icol(n)-3
       sfsgs3(n)=sfsgs3(n)+fsgs3(i)
170    continue
      if(ic(n).le.4)goto 200
      do 180 i=icol(n-1)+1,icol(n)-4
```

```
        sfsgs4(n)=sfsgs4(n)+fsgs4(i)
 180     continue
       if(ic(n).le.5)goto 200
       do 190 i=icol(n-1)+1,icol(n)-5
        sfsgs5(n)=sfsgs5(n)+fsgs5(i)
 190     continue
 200     sfsgg(n)=sfsgs0(n)+sfsgs1(n)+sfsgs2(n)+sfsgs3(n)+
   +            sfsgs4(n)+sfsgs5(n)
      endif
  90  continue
*****************************************************
*************************
* Calculate the numerator for the broad group down scatter g-g+1.
*
     if(nbg.eq.1)goto 260
     do 210 n=1,nbg-1
      sfsgs11(n)=fsgs1(icol(n))
      if(ic(n).eq.1)then
       if(ic(n+1).eq.1)then
        sfsgs21(n)=0.0
        sfsgs31(n)=0.0
        sfsgs41(n)=0.0
        sfsgs51(n)=0.0
       elseif(ic(n+1).eq.2)then
        sfsgs21(n)=fsgs2(icol(n))
        sfsgs31(n)=0.0
        sfsgs41(n)=0.0
        sfsgs51(n)=0.0
       elseif(ic(n+1).eq.3)then
        sfsgs21(n)=fsgs2(icol(n))
        sfsgs31(n)=fsgs3(icol(n))
        sfsgs41(n)=0.0
        sfsgs51(n)=0.0
       elseif(ic(n+1).eq.4)then
        sfsgs21(n)=fsgs2(icol(n))
        sfsgs31(n)=fsgs3(icol(n))
        sfsgs41(n)=fsgs4(icol(n))
        sfsgs51(n)=0.0
       elseif(ic(n+1).ge.5)then
        sfsgs21(n)=fsgs2(icol(n))
```

```
    sfsgs31(n)=fsgs3(icol(n))
    sfsgs41(n)=fsgs4(icol(n))
    sfsgs51(n)=fsgs5(icol(n))
   endif
 elseif(ic(n).eq.2)then
   if(ic(n+1).eq.1)then
    sfsgs21(n)=fsgs2(icol(n)-1)
    sfsgs31(n)=0.0
    sfsgs41(n)=0.0
    sfsgs51(n)=0.0
   elseif(ic(n+1).eq.2)then
    sfsgs21(n)=fsgs2(icol(n)-1)+fsgs2(icol(n))
    sfsgs31(n)=fsgs3(icol(n)-1)
    sfsgs41(n)=0.0
    sfsgs51(n)=0.0
   elseif(ic(n+1).eq.3)then
    sfsgs21(n)=fsgs2(icol(n)-1)+fsgs2(icol(n))
    sfsgs31(n)=fsgs3(icol(n)-1)+fsgs3(icol(n))
    sfsgs41(n)=fsgs4(icol(n)-1)
    sfsgs51(n)=0.0
   elseif(ic(n+1).eq.4)then
    sfsgs21(n)=fsgs2(icol(n)-1)+fsgs2(icol(n))
    sfsgs31(n)=fsgs3(icol(n)-1)+fsgs3(icol(n))
    sfsgs41(n)=fsgs4(icol(n)-1)+fsgs4(icol(n))
    sfsgs51(n)=fsgs5(icol(n)-1)
   elseif(ic(n+1).ge.5)then
    sfsgs21(n)=fsgs2(icol(n)-1)+fsgs2(icol(n))
    sfsgs31(n)=fsgs3(icol(n)-1)+fsgs3(icol(n))
    sfsgs41(n)=fsgs4(icol(n)-1)+fsgs4(icol(n))
    sfsgs51(n)=fsgs5(icol(n)-1)+fsgs5(icol(n))
   endif
 elseif(ic(n).eq.3)then
   if(ic(n+1).eq.1)then
    sfsgs21(n)=fsgs2(icol(n)-1)
    sfsgs31(n)=fsgs3(icol(n)-2)
    sfsgs41(n)=0.0
    sfsgs51(n)=0.0
   elseif(ic(n+1).eq.2)then
    sfsgs21(n)=fsgs2(icol(n)-1)+fsgs2(icol(n))
    sfsgs31(n)=fsgs3(icol(n)-2)+fsgs3(icol(n)-1)
```

```
      sfsgs41(n)=fsgs4(icol(n)-2)
      sfsgs51(n)=0.0
     elseif(ic(n+1).eq.3)then
      sfsgs21(n)=fsgs2(icol(n)-1)+fsgs2(icol(n))
      sfsgs31(n)=fsgs3(icol(n)-2)+fsgs3(icol(n)-1)+fsgs3(icol(n))
      sfsgs41(n)=fsgs4(icol(n)-2)+fsgs4(icol(n)-1)
      sfsgs51(n)=fsgs5(icol(n)-2)
     elseif(ic(n+1).eq.4)then
      sfsgs21(n)=fsgs2(icol(n)-1)+fsgs2(icol(n))
      sfsgs31(n)=fsgs3(icol(n)-2)+fsgs3(icol(n)-1)+fsgs3(icol(n))
      sfsgs41(n)=fsgs4(icol(n)-2)+fsgs4(icol(n)-1)+fsgs4(icol(n))
      sfsgs51(n)=fsgs5(icol(n)-2)+fsgs5(icol(n)-1)
     elseif(ic(n+1).ge.5)then
      sfsgs21(n)=fsgs2(icol(n)-1)+fsgs2(icol(n))
      sfsgs31(n)=fsgs3(icol(n)-2)+fsgs3(icol(n)-1)+fsgs3(icol(n))
      sfsgs41(n)=fsgs4(icol(n)-2)+fsgs4(icol(n)-1)+fsgs4(icol(n))
      sfsgs51(n)=fsgs5(icol(n)-2)+fsgs5(icol(n)-1)+fsgs5(icol(n))
     endif
    elseif(ic(n).eq.4)then
     if(ic(n+1).eq.1)then
      sfsgs21(n)=fsgs2(icol(n)-1)
      sfsgs31(n)=fsgs3(icol(n)-2)
      sfsgs41(n)=fsgs4(icol(n)-3)
      sfsgs51(n)=0.0
     elseif(ic(n+1).eq.2)then
      sfsgs21(n)=fsgs2(icol(n)-1)+fsgs2(icol(n))
      sfsgs31(n)=fsgs3(icol(n)-2)+fsgs3(icol(n)-1)
      sfsgs41(n)=fsgs4(icol(n)-3)+fsgs4(icol(n)-2)
      sfsgs51(n)=fsgs5(icol(n)-3)
     elseif(ic(n+1).eq.3)then
      sfsgs21(n)=fsgs2(icol(n)-1)+fsgs2(icol(n))
      sfsgs31(n)=fsgs3(icol(n)-2)+fsgs3(icol(n)-1)+fsgs3(icol(n))
      sfsgs41(n)=fsgs4(icol(n)-3)+fsgs4(icol(n)-2)+
 +             fsgs4(icol(n)-1)
      sfsgs51(n)=fsgs5(icol(n)-3)+fsgs5(icol(n)-2)
     elseif(ic(n+1).eq.4)then
      sfsgs21(n)=fsgs2(icol(n)-1)+fsgs2(icol(n))
      sfsgs31(n)=fsgs3(icol(n)-2)+fsgs3(icol(n)-1)+fsgs3(icol(n))
      sfsgs41(n)=fsgs4(icol(n)-3)+fsgs4(icol(n)-2)+
 +             fsgs4(icol(n)-1)+fsgs4(icol(n))
```

```
      sfsgs51(n)=fsgs5(icol(n)-3)+fsgs5(icol(n)-2)+
+           fsgs5(icol(n)-1)
     elseif(ic(n+1).ge.5)then
      sfsgs21(n)=fsgs2(icol(n)-1)+fsgs2(icol(n))
      sfsgs31(n)=fsgs3(icol(n)-2)+fsgs3(icol(n)-1)+fsgs3(icol(n))
      sfsgs41(n)=fsgs4(icol(n)-3)+fsgs4(icol(n)-2)+
+           fsgs4(icol(n)-1)+fsgs4(icol(n))
      sfsgs51(n)=fsgs5(icol(n)-3)+fsgs5(icol(n)-2)+
+           fsgs5(icol(n)-1)+fsgs5(icol(n))
     endif
    elseif(ic(n).ge.5)then
     if(ic(n+1).eq.1)then
      sfsgs21(n)=fsgs2(icol(n)-1)
      sfsgs31(n)=fsgs3(icol(n)-2)
      sfsgs41(n)=fsgs4(icol(n)-3)
      sfsgs51(n)=fsgs5(icol(n)-4)
     elseif(ic(n+1).eq.2)then
      sfsgs21(n)=fsgs2(icol(n)-1)+fsgs2(icol(n))
      sfsgs31(n)=fsgs3(icol(n)-2)+fsgs3(icol(n)-1)
      sfsgs41(n)=fsgs4(icol(n)-3)+fsgs4(icol(n)-2)
      sfsgs51(n)=fsgs5(icol(n)-4)+fsgs5(icol(n)-3)
     elseif(ic(n+1).eq.3)then
      sfsgs21(n)=fsgs2(icol(n)-1)+fsgs2(icol(n))
      sfsgs31(n)=fsgs3(icol(n)-2)+fsgs3(icol(n)-1)+fsgs3(icol(n))
      sfsgs41(n)=fsgs4(icol(n)-3)+fsgs4(icol(n)-2)+
+           fsgs4(icol(n)-1)
      sfsgs51(n)=fsgs5(icol(n)-4)+fsgs5(icol(n)-3)+
+           fsgs5(icol(n)-2)
     elseif(ic(n+1).eq.4)then
      sfsgs21(n)=fsgs2(icol(n)-1)+fsgs2(icol(n))
      sfsgs31(n)=fsgs3(icol(n)-2)+fsgs3(icol(n)-1)+fsgs3(icol(n))
      sfsgs41(n)=fsgs4(icol(n)-3)+fsgs4(icol(n)-2)+
+           fsgs4(icol(n)-1)+fsgs4(icol(n))
      sfsgs51(n)=fsgs5(icol(n)-4)+fsgs5(icol(n)-3)+
+           fsgs5(icol(n)-2)+fsgs5(icol(n)-1)
     elseif(ic(n+1).ge.5)then
      sfsgs21(n)=fsgs2(icol(n)-1)+fsgs2(icol(n))
      sfsgs31(n)=fsgs3(icol(n)-2)+fsgs3(icol(n)-1)+fsgs3(icol(n))
      sfsgs41(n)=fsgs4(icol(n)-3)+fsgs4(icol(n)-2)+
+           fsgs4(icol(n)-1)+fsgs4(icol(n))
```

```
      sfsgs51(n)=fsgs5(icol(n)-4)+fsgs5(icol(n)-3)+
   +            fsgs5(icol(n)-2)+fsgs5(icol(n)-1)+fsgs5(icol(n))
     endif
    endif
    sfsgg1(n)=sfsgs11(n)+sfsgs21(n)+sfsgs31(n)+sfsgs41(n)+
   +          sfsgs51(n)
 210 continue
*****************************************************
************************
* Calculate numerators for broad group down scatter g-g+2
*
    if(nbg.le.2)goto 260
    do 220 n=1,nbg-2
     if(ic(n).eq.1)then
      if(ic(n+1).eq.1)then
       if(ic(n+2).eq.1)then
        sfsgs22(n)=fsgs2(icol(n))
        sfsgs32(n)=0.0
        sfsgs42(n)=0.0
        sfsgs52(n)=0.0
       elseif(ic(n+2).eq.2)then
        sfsgs22(n)=fsgs2(icol(n))
        sfsgs32(n)=fsgs3(icol(n))
        sfsgs42(n)=0.0
        sfsgs52(n)=0.0
       elseif(ic(n+2).eq.3)then
        sfsgs22(n)=fsgs2(icol(n))
        sfsgs32(n)=fsgs3(icol(n))
        sfsgs42(n)=fsgs4(icol(n))
        sfsgs52(n)=0.0
       elseif(ic(n+2).ge.4)then
        sfsgs22(n)=fsgs2(icol(n))
        sfsgs32(n)=fsgs3(icol(n))
        sfsgs42(n)=fsgs4(icol(n))
        sfsgs52(n)=fsgs5(icol(n))
       endif
      elseif(ic(n+1).eq.2)then
       if(ic(n+2).eq.1)then
        sfsgs22(n)=0.0
        sfsgs32(n)=fsgs3(icol(n))
```

```
      sfsgs42(n)=0.0
      sfsgs52(n)=0.0
    elseif(ic(n+2).eq.2)then
      sfsgs22(n)=0.0
      sfsgs32(n)=fsgs3(icol(n))
      sfsgs42(n)=fsgs4(icol(n))
      sfsgs52(n)=0.0
    elseif(ic(n+2).ge.3)then
      sfsgs22(n)=0.0
      sfsgs32(n)=fsgs3(icol(n))
      sfsgs42(n)=fsgs4(icol(n))
      sfsgs52(n)=fsgs5(icol(n))
    endif
  elseif(ic(n+1).eq.3)then
    if(ic(n+2).eq.1)then
      sfsgs22(n)=0.0
      sfsgs32(n)=0.0
      sfsgs42(n)=fsgs4(icol(n))
      sfsgs52(n)=0.0
    elseif(ic(n+2).ge.2)then
      sfsgs22(n)=0.0
      sfsgs32(n)=0.0
      sfsgs42(n)=fsgs4(icol(n))
      sfsgs52(n)=fsgs5(icol(n))
    endif
  elseif(ic(n+1).eq.4)then
    sfsgs22(n)=0.0
    sfsgs32(n)=0.0
    sfsgs42(n)=0.0
    sfsgs52(n)=fsgs5(icol(n))
  elseif(ic(n+1).ge.5)then
    sfsgs22(n)=0.0
    sfsgs32(n)=0.0
    sfsgs42(n)=0.0
    sfsgs52(n)=0.0
  endif
elseif(ic(n).eq.2)then
  if(ic(n+1).eq.1)then
    if(ic(n+2).eq.1)then
      sfsgs22(n)=fsgs2(icol(n))
```

```
    sfsgs32(n)=fsgs3(icol(n)-1)
    sfsgs42(n)=0.0
    sfsgs52(n)=0.0
   elseif(ic(n+2).eq.2)then
    sfsgs22(n)=fsgs2(icol(n))
    sfsgs32(n)=fsgs3(icol(n)-1)+fsgs3(icol(n))
    sfsgs42(n)=fsgs4(icol(n)-1)
    sfsgs52(n)=0.0
   elseif(ic(n+2).eq.3)then
    sfsgs22(n)=fsgs2(icol(n))
    sfsgs32(n)=fsgs3(icol(n)-1)+fsgs3(icol(n))
    sfsgs42(n)=fsgs4(icol(n)-1)+fsgs4(icol(n))
    sfsgs52(n)=fsgs5(icol(n)-1)
   elseif(ic(n+2).ge.4)then
    sfsgs22(n)=fsgs2(icol(n))
    sfsgs32(n)=fsgs3(icol(n)-1)+fsgs3(icol(n))
    sfsgs42(n)=fsgs4(icol(n)-1)+fsgs4(icol(n))
    sfsgs52(n)=fsgs5(icol(n)-1)+fsgs5(icol(n))
   endif
  elseif(ic(n+1).eq.2)then
   if(ic(n+2).eq.1)then
    sfsgs22(n)=0.0
    sfsgs32(n)=fsgs3(icol(n))
    sfsgs42(n)=fsgs4(icol(n)-1)
    sfsgs52(n)=0.0
   elseif(ic(n+2).eq.2)then
    sfsgs22(n)=0.0
    sfsgs32(n)=fsgs3(icol(n))
    sfsgs42(n)=fsgs4(icol(n)-1)+fsgs4(icol(n))
    sfsgs52(n)=fsgs5(icol(n)-1)
   elseif(ic(n+2).ge.3)then
    sfsgs22(n)=0.0
    sfsgs32(n)=fsgs3(icol(n))
    sfsgs42(n)=fsgs4(icol(n)-1)+fsgs4(icol(n))
    sfsgs52(n)=fsgs5(icol(n)-1)+fsgs5(icol(n))
   endif
  elseif(ic(n+1).eq.3)then
   if(ic(n+2).eq.1)then
    sfsgs22(n)=0.0
    sfsgs32(n)=0.0
```

```
      sfsgs42(n)=fsgs4(icol(n))
      sfsgs52(n)=fsgs5(icol(n)-1)
     elseif(ic(n+2).ge.2)then
      sfsgs22(n)=0.0
      sfsgs32(n)=0.0
      sfsgs42(n)=fsgs4(icol(n))
      sfsgs52(n)=fsgs5(icol(n)-1)+fsgs5(icol(n))
     endif
    elseif(ic(n+1).eq.4)then
     sfsgs22(n)=0.0
     sfsgs32(n)=0.0
     sfsgs42(n)=0.0
     sfsgs52(n)=fsgs5(icol(n))
    elseif(ic(n+1).ge.5)then
     sfsgs22(n)=0.0
     sfsgs32(n)=0.0
     sfsgs42(n)=0.0
     sfsgs52(n)=0.0
    endif
   elseif(ic(n).eq.3)then
    if(ic(n+1).eq.1)then
     if(ic(n+2).eq.1)then
      sfsgs22(n)=fsgs2(icol(n))
      sfsgs32(n)=fsgs3(icol(n)-1)
      sfsgs42(n)=fsgs4(icol(n)-2)
      sfsgs52(n)=0.0
     elseif(ic(n+2).eq.2)then
      sfsgs22(n)=fsgs2(icol(n))
      sfsgs32(n)=fsgs3(icol(n)-1)+fsgs3(icol(n))
      sfsgs42(n)=fsgs4(icol(n)-2)+fsgs4(icol(n)-1)
      sfsgs52(n)=fsgs5(icol(n)-2)
     elseif(ic(n+2).eq.3)then
      sfsgs22(n)=fsgs2(icol(n))
      sfsgs32(n)=fsgs3(icol(n)-1)+fsgs3(icol(n))
      sfsgs42(n)=fsgs4(icol(n)-2)+fsgs4(icol(n)-1)+
+                fsgs4(icol(n))
      sfsgs52(n)=fsgs5(icol(n)-2)+fsgs5(icol(n)-1)
     elseif(ic(n+2).ge.4)then
      sfsgs22(n)=fsgs2(icol(n))
      sfsgs32(n)=fsgs3(icol(n)-1)+fsgs3(icol(n))
```

```
        sfsgs42(n)=fsgs4(icol(n)-2)+fsgs4(icol(n)-1)+
     +          fsgs4(icol(n))
        sfsgs52(n)=fsgs5(icol(n)-2)+fsgs5(icol(n)-1)+
     +          fsgs5(icol(n))
       endif
      elseif(ic(n+1).eq.2)then
       if(ic(n+2).eq.1)then
        sfsgs22(n)=0.0
        sfsgs32(n)=fsgs3(icol(n))
        sfsgs42(n)=fsgs4(icol(n)-1)
        sfsgs52(n)=fsgs5(icol(n)-2)
       elseif(ic(n+2).eq.2)then
        sfsgs22(n)=0.0
        sfsgs32(n)=fsgs3(icol(n))
        sfsgs42(n)=fsgs4(icol(n)-1)+fsgs4(icol(n))
        sfsgs52(n)=fsgs5(icol(n)-2)+fsgs5(icol(n)-1)
       elseif(ic(n+2).ge.3)then
        sfsgs22(n)=0.0
        sfsgs32(n)=fsgs3(icol(n))
        sfsgs42(n)=fsgs4(icol(n)-1)+fsgs4(icol(n))
        sfsgs52(n)=fsgs5(icol(n)-2)+fsgs5(icol(n)-1)+
     +          fsgs5(icol(n))
       endif
      elseif(ic(n+1).eq.3)then
       if(ic(n+2).eq.1)then
        sfsgs22(n)=0.0
        sfsgs32(n)=0.0
        sfsgs42(n)=fsgs4(icol(n))
        sfsgs52(n)=fsgs5(icol(n)-1)
       elseif(ic(n+2).ge.2)then
        sfsgs22(n)=0.0
        sfsgs32(n)=0.0
        sfsgs42(n)=fsgs4(icol(n))
        sfsgs52(n)=fsgs5(icol(n)-1)+fsgs5(icol(n))
       endif
      elseif(ic(n+1).eq.4)then
       sfsgs22(n)=0.0
       sfsgs32(n)=0.0
       sfsgs42(n)=0.0
       sfsgs52(n)=fsgs5(icol(n))
```

```
        elseif(ic(n+1).ge.5)then
          sfsgs22(n)=0.0
          sfsgs32(n)=0.0
          sfsgs42(n)=0.0
          sfsgs52(n)=0.0
        endif
      elseif(ic(n).ge.4)then
        if(ic(n+1).eq.1)then
          if(ic(n+2).eq.1)then
            sfsgs22(n)=fsgs2(icol(n))
            sfsgs32(n)=fsgs3(icol(n)-1)
            sfsgs42(n)=fsgs4(icol(n)-2)
            sfsgs52(n)=fsgs5(icol(n)-3)
          elseif(ic(n+2).eq.2)then
            sfsgs22(n)=fsgs2(icol(n))
            sfsgs32(n)=fsgs3(icol(n)-1)+fsgs3(icol(n))
            sfsgs42(n)=fsgs4(icol(n)-2)+fsgs4(icol(n)-1)
            sfsgs52(n)=fsgs5(icol(n)-3)+fsgs5(icol(n)-2)
          elseif(ic(n+2).eq.3)then
            sfsgs22(n)=fsgs2(icol(n))
            sfsgs32(n)=fsgs3(icol(n)-1)+fsgs3(icol(n))
            sfsgs42(n)=fsgs4(icol(n)-2)+fsgs4(icol(n)-1)+
     +              fsgs4(icol(n))
            sfsgs52(n)=fsgs5(icol(n)-3)+fsgs5(icol(n)-2)+
     +              fsgs5(icol(n)-1)
          elseif(ic(n+2).ge.4)then
            sfsgs22(n)=fsgs2(icol(n))
            sfsgs32(n)=fsgs3(icol(n)-1)+fsgs3(icol(n))
            sfsgs42(n)=fsgs4(icol(n)-2)+fsgs4(icol(n)-1)+
     +              fsgs4(icol(n))
            sfsgs52(n)=fsgs5(icol(n)-3)+fsgs5(icol(n)-2)+
     +              fsgs5(icol(n)-1)+fsgs5(icol(n))
          endif
        elseif(ic(n+1).eq.2)then
          if(ic(n+2).eq.1)then
            sfsgs22(n)=0.0
            sfsgs32(n)=fsgs3(icol(n))
            sfsgs42(n)=fsgs4(icol(n)-1)
            sfsgs52(n)=fsgs5(icol(n)-2)
          elseif(ic(n+2).eq.2)then
```

```
          sfsgs22(n)=0.0
          sfsgs32(n)=fsgs3(icol(n))
          sfsgs42(n)=fsgs4(icol(n)-1)+fsgs4(icol(n))
          sfsgs52(n)=fsgs5(icol(n)-2)+fsgs5(icol(n)-1)
         elseif(ic(n+2).ge.3)then
          sfsgs22(n)=0.0
          sfsgs32(n)=fsgs3(icol(n))
          sfsgs42(n)=fsgs4(icol(n)-1)+fsgs4(icol(n))
          sfsgs52(n)=fsgs5(icol(n)-2)+fsgs5(icol(n)-1)+
     +              fsgs5(icol(n))
         endif
        elseif(ic(n+1).eq.3)then
         if(ic(n+2).eq.1)then
          sfsgs22(n)=0.0
          sfsgs32(n)=0.0
          sfsgs42(n)=fsgs4(icol(n))
          sfsgs52(n)=fsgs5(icol(n)-1)
         elseif(ic(n+2).ge.2)then
          sfsgs22(n)=0.0
          sfsgs32(n)=0.0
          sfsgs42(n)=fsgs4(icol(n))
          sfsgs52(n)=fsgs5(icol(n)-1)+fsgs5(icol(n))
         endif
        elseif(ic(n+1).eq.4)then
         sfsgs22(n)=0.0
         sfsgs32(n)=0.0
         sfsgs42(n)=0.0
         sfsgs52(n)=fsgs5(icol(n))
        elseif(ic(n+1).ge.5)then
         sfsgs22(n)=0.0
         sfsgs32(n)=0.0
         sfsgs42(n)=0.0
         sfsgs52(n)=0.0
        endif
       endif
       sfsgg2(n)=sfsgs22(n)+sfsgs32(n)+sfsgs42(n)+sfsgs52(n)
 220  continue
*********************************************************
********************
* Calculate numerators for broad group down scatter g-g+3.
```

```
*
      if(nbg.le.3)goto 260
      do 230 n=1,nbg-3
       if(ic(n).eq.1)then
        if(ic(n+1).eq.1)then
         if(ic(n+2).eq.1)then
          if(ic(n+3).eq.1)then
           sfsgs33(n)=fsgs3(icol(n))
           sfsgs43(n)=0.0
           sfsgs53(n)=0.0
          elseif(ic(n+3).eq.2)then
           sfsgs33(n)=fsgs3(icol(n))
           sfsgs43(n)=fsgs4(icol(n))
           sfsgs53(n)=0.0
          elseif(ic(n+3).ge.3)then
           sfsgs33(n)=fsgs3(icol(n))
           sfsgs43(n)=fsgs4(icol(n))
           sfsgs53(n)=fsgs5(icol(n))
          endif
         elseif(ic(n+2).eq.2)then
          if(ic(n+3).eq.1)then
           sfsgs33(n)=0.0
           sfsgs43(n)=fsgs4(icol(n))
           sfsgs53(n)=0.0
          elseif(ic(n+3).ge.2)then
           sfsgs33(n)=0.0
           sfsgs43(n)=fsgs4(icol(n))
           sfsgs53(n)=fsgs5(icol(n))
          endif
         elseif(ic(n+2).eq.3)then
          sfsgs33(n)=0.0
          sfsgs43(n)=0.0
          sfsgs53(n)=fsgs5(icol(n))
         elseif(ic(n+2).ge.4)then
          sfsgs33(n)=0.0
          sfsgs43(n)=0.0
          sfsgs53(n)=0.0
         endif
        elseif(ic(n+1).eq.2)then
         if(ic(n+2).eq.1)then
```

```
      if(ic(n+3).eq.1)then
        sfsgs33(n)=0.0
        sfsgs43(n)=fsgs4(icol(n))
        sfsgs53(n)=0.0
      elseif(ic(n+3).ge.2)then
        sfsgs33(n)=0.0
        sfsgs43(n)=fsgs4(icol(n))
        sfsgs53(n)=fsgs5(icol(n))
      endif
    elseif(ic(n+2).eq.2)then
      sfsgs33(n)=0.0
      sfsgs43(n)=0.0
      sfsgs53(n)=fsgs5(icol(n))
    elseif(ic(n+2).ge.3)then
      sfsgs33(n)=0.0
      sfsgs43(n)=0.0
      sfsgs53(n)=0.0
    endif
  elseif(ic(n+1).eq.3)then
    if(ic(n+2).eq.1)then
      sfsgs33(n)=0.0
      sfsgs43(n)=0.0
      sfsgs53(n)=fsgs5(icol(n))
    elseif(ic(n+2).ge.2)then
      sfsgs33(n)=0.0
      sfsgs43(n)=0.0
      sfsgs53(n)=0.0
    endif
  elseif(ic(n+1).ge.4)then
    sfsgs33(n)=0.0
    sfsgs43(n)=0.0
    sfsgs53(n)=0.0
  endif
elseif(ic(n).eq.2)then
  if(ic(n+1).eq.1)then
    if(ic(n+2).eq.1)then
      if(ic(n+3).eq.1)then
        sfsgs33(n)=fsgs3(icol(n))
        sfsgs43(n)=fsgs4(icol(n)-1)
        sfsgs53(n)=0.0
```

```
      elseif(ic(n+3).eq.2)then
        sfsgs33(n)=fsgs3(icol(n))
        sfsgs43(n)=fsgs4(icol(n)-1)+fsgs4(icol(n))
        sfsgs53(n)=fsgs5(icol(n)-1)
      elseif(ic(n+3).ge.3)then
        sfsgs33(n)=fsgs3(icol(n))
        sfsgs43(n)=fsgs4(icol(n)-1)+fsgs4(icol(n))
        sfsgs53(n)=fsgs5(icol(n)-1)+fsgs5(icol(n))
      endif
    elseif(ic(n+2).eq.2)then
      if(ic(n+3).eq.1)then
        sfsgs33(n)=0.0
        sfsgs43(n)=fsgs4(icol(n))
        sfsgs53(n)=fsgs5(icol(n)-1)
      elseif(ic(n+3).ge.2)then
        sfsgs33(n)=0.0
        sfsgs43(n)=fsgs4(icol(n))
        sfsgs53(n)=fsgs5(icol(n)-1)+fsgs5(icol(n))
      endif
    elseif(ic(n+2).eq.3)then
      sfsgs33(n)=0.0
      sfsgs43(n)=0.0
      sfsgs53(n)=fsgs5(icol(n))
    elseif(ic(n+2).ge.4)then
      sfsgs33(n)=0.0
      sfsgs43(n)=0.0
      sfsgs53(n)=0.0
    endif
  elseif(ic(n+1).eq.2)then
    if(ic(n+2).eq.1)then
      if(ic(n+3).eq.1)then
        sfsgs33(n)=0.0
        sfsgs43(n)=fsgs4(icol(n))
        sfsgs53(n)=fsgs5(icol(n)-1)
      elseif(ic(n+3).ge.2)then
        sfsgs33(n)=0.0
        sfsgs43(n)=fsgs4(icol(n))
        sfsgs53(n)=fsgs5(icol(n)-1)+fsgs5(icol(n))
      endif
    elseif(ic(n+2).eq.2)then
```

```
        sfsgs33(n)=0.0
        sfsgs43(n)=0.0
        sfsgs53(n)=fsgs5(icol(n))
      elseif(ic(n+2).ge.3)then
        sfsgs33(n)=0.0
        sfsgs43(n)=0.0
        sfsgs53(n)=0.0
      endif
    elseif(ic(n+1).eq.3)then
      if(ic(n+2).eq.1)then
        sfsgs33(n)=0.0
        sfsgs43(n)=0.0
        sfsgs53(n)=fsgs5(icol(n))
      elseif(ic(n+2).ge.2)then
        sfsgs33(n)=0.0
        sfsgs43(n)=0.0
        sfsgs53(n)=0.0
      endif
    elseif(ic(n+1).ge.4)then
      sfsgs33(n)=0.0
      sfsgs43(n)=0.0
      sfsgs53(n)=0.0
    endif
  elseif(ic(n).ge.3)then
    if(ic(n+1).eq.1)then
      if(ic(n+2).eq.1)then
        if(ic(n+3).eq.1)then
          sfsgs33(n)=fsgs3(icol(n))
          sfsgs43(n)=fsgs4(icol(n)-1)
          sfsgs53(n)=fsgs5(icol(n)-2)
        elseif(ic(n+3).eq.2)then
          sfsgs33(n)=fsgs3(icol(n))
          sfsgs43(n)=fsgs4(icol(n)-1)+fsgs4(icol(n))
          sfsgs53(n)=fsgs5(icol(n)-2)+fsgs5(icol(n)-1)
        elseif(ic(n+3).ge.3)then
          sfsgs33(n)=fsgs3(icol(n))
          sfsgs43(n)=fsgs4(icol(n)-1)+fsgs4(icol(n))
          sfsgs53(n)=fsgs5(icol(n)-2)+fsgs5(icol(n)-1)+
+                    fsgs5(icol(n))
        endif
```

```
  elseif(ic(n+2).eq.2)then
    if(ic(n+3).eq.1)then
      sfsgs33(n)=0.0
      sfsgs43(n)=fsgs4(icol(n))
      sfsgs53(n)=fsgs5(icol(n)-1)
    elseif(ic(n+3).ge.2)then
      sfsgs33(n)=0.0
      sfsgs43(n)=fsgs4(icol(n))
      sfsgs53(n)=fsgs5(icol(n)-1)+fsgs5(icol(n))
    endif
  elseif(ic(n+2).eq.3)then
    sfsgs33(n)=0.0
    sfsgs43(n)=0.0
    sfsgs53(n)=fsgs5(icol(n))
  elseif(ic(n+2).ge.4)then
    sfsgs33(n)=0.0
    sfsgs43(n)=0.0
    sfsgs53(n)=0.0
  endif
elseif(ic(n+1).eq.2)then
  if(ic(n+2).eq.1)then
    if(ic(n+3).eq.1)then
      sfsgs33(n)=0.0
      sfsgs43(n)=fsgs4(icol(n))
      sfsgs53(n)=fsgs5(icol(n)-1)
    elseif(ic(n+3).ge.2)then
      sfsgs33(n)=0.0
      sfsgs43(n)=fsgs4(icol(n))
      sfsgs53(n)=fsgs5(icol(n)-1)+fsgs5(icol(n))
    endif
  elseif(ic(n+2).eq.2)then
    sfsgs33(n)=0.0
    sfsgs43(n)=0.0
    sfsgs53(n)=fsgs5(icol(n))
  elseif(ic(n+2).ge.3)then
    sfsgs33(n)=0.0
    sfsgs43(n)=0.0
    sfsgs53(n)=0.0
  endif
elseif(ic(n+1).eq.3)then
```

```
        if(ic(n+2).eq.1)then
          sfsgs33(n)=0.0
          sfsgs43(n)=0.0
          sfsgs53(n)=fsgs5(icol(n))
        elseif(ic(n+2).ge.2)then
          sfsgs33(n)=0.0
          sfsgs43(n)=0.0
          sfsgs53(n)=0.0
        endif
      elseif(ic(n+1).ge.4)then
        sfsgs33(n)=0.0
        sfsgs43(n)=0.0
        sfsgs53(n)=0.0
      endif
    endif
    sfsgg3(n)=sfsgs33(n)+sfsgs43(n)+sfsgs53(n)
230 continue
************************************************************
*************************
* Calculate the numerators for broad group down scatter g-g+4
*
   if(nbg.le.4)goto 260
   do 240 n=1,nbg-4
    if(ic(n).eq.1)then
      if(ic(n+1).eq.1)then
        if(ic(n+2).eq.1)then
          if(ic(n+3).eq.1)then
            if(ic(n+4).eq.1)then
              sfsgs44(n)=fsgs4(icol(n))
              sfsgs54(n)=0.0
            elseif(ic(n+4).ge.2)then
              sfsgs44(n)=fsgs4(icol(n))
              sfsgs54(n)=fsgs5(icol(n))
            endif
          elseif(ic(n+3).eq.2)then
            sfsgs44(n)=0.0
            sfsgs54(n)=fsgs5(icol(n))
          elseif(ic(n+3).ge.3)then
            sfsgs44(n)=0.0
            sfsgs54(n)=0.0
```

```
        endif
      elseif(ic(n+2).eq.2)then
        if(ic(n+3).eq.1)then
          sfsgs44(n)=0.0
          sfsgs54(n)=fsgs5(icol(n))
        elseif(ic(n+3).ge.2)then
          sfsgs44(n)=0.0
          sfsgs54(n)=0.0
        endif
      elseif(ic(n+2).ge.3)then
        sfsgs44(n)=0.0
        sfsgs54(n)=0.0
      endif
    elseif(ic(n+1).eq.2)then
      if(ic(n+2).eq.1)then
        if(ic(n+3).eq.1)then
          sfsgs44(n)=0.0
          sfsgs54(n)=fsgs5(icol(n))
        elseif(ic(n+3).ge.2)then
          sfsgs44(n)=0.0
          sfsgs54(n)=0.0
        endif
      elseif(ic(n+2).ge.2)then
        sfsgs44(n)=0.0
        sfsgs54(n)=0.0
      endif
    elseif(ic(n+1).ge.3)then
      sfsgs44(n)=0.0
      sfsgs54(n)=0.0
    endif
  elseif(ic(n).ge.2)then
    if(ic(n+1).eq.1)then
      if(ic(n+2).eq.1)then
        if(ic(n+3).eq.1)then
          if(ic(n+4).eq.1)then
            sfsgs44(n)=fsgs4(icol(n))
            sfsgs54(n)=fsgs5(icol(n)-1)
          elseif(ic(n+4).ge.2)then
            sfsgs44(n)=fsgs4(icol(n))
            sfsgs54(n)=fsgs5(icol(n)-1)+fsgs5(icol(n))
```

```
          endif
        elseif(ic(n+3).eq.2)then
          sfsgs44(n)=0.0
          sfsgs54(n)=fsgs5(icol(n))
        elseif(ic(n+3).ge.3)then
          sfsgs44(n)=0.0
          sfsgs54(n)=0.0
        endif
      elseif(ic(n+2).eq.2)then
        if(ic(n+3).eq.1)then
          sfsgs44(n)=0.0
          sfsgs54(n)=fsgs5(icol(n))
        elseif(ic(n+3).ge.2)then
          sfsgs44(n)=0.0
          sfsgs54(n)=0.0
        endif
      elseif(ic(n+2).ge.3)then
        sfsgs44(n)=0.0
        sfsgs54(n)=0.0
      endif
    elseif(ic(n+1).eq.2)then
      if(ic(n+2).eq.1)then
        if(ic(n+3).eq.1)then
          sfsgs44(n)=0.0
          sfsgs54(n)=fsgs5(icol(n))
        elseif(ic(n+3).ge.2)then
          sfsgs44(n)=0.0
          sfsgs54(n)=0.0
        endif
      elseif(ic(n+2).ge.2)then
        sfsgs44(n)=0.0
        sfsgs54(n)=0.0
      endif
    elseif(ic(n+1).ge.3)then
      sfsgs44(n)=0.0
      sfsgs54(n)=0.0
    endif
  endif
  sfsgg4(n)=sfsgs44(n)+sfsgs54(n)
240 continue
```

```
*******************************************************
*************************
* Calculate numerators for broad group down scatter g-g+5
*
      if(nbg.le.5)goto 260
      do 250 n=1,nbg-5
       if(ic(n+1).eq.1)then
        if(ic(n+2).eq.1)then
         if(ic(n+3).eq.1)then
          if(ic(n+4).eq.1)then
           sfsgs55(n)=fsgs5(icol(n))
          else
           sfsgs55(n)=0.0
          endif
         else
          sfsgs55(n)=0.0
         endif
        else
         sfsgs55(n)=0.0
        endif
       else
        sfsgs55(n)=0.0
       endif
 250  continue
*******************************************************
*************************
* Calculate cross sections for each broad group.
*
 260  do 270 n=1,nbg
       nusigf(n)=sfnusf(n)/sflux(n)
       sigt(n)=sfsigt(n)/sflux(n)
       siga(n)=sfsiga(n)/sflux(n)
       sgsgg(n)=sfsgg(n)/sflux(n)
       sgsgg1(n)=sfsgg1(n)/sflux(n)
       sgsgg2(n)=sfsgg2(n)/sflux(n)
       sgsgg3(n)=sfsgg3(n)/sflux(n)
       sgsgg4(n)=sfsgg4(n)/sflux(n)
       sgsgg5(n)=sfsgs55(n)/sflux(n)
 270  continue
* For converting micro to macro cross sections.
```

```
      if(adh.ne.0)then
       do 275 n=1,nbg
        nusigf(n)=nusigf(n)*adh
        sigt(n)=sigt(n)*adh
        siga(n)=siga(n)*adh
        sgsgg(n)=sgsgg(n)*adh
        sgsgg1(n)=sgsgg1(n)*adh
        sgsgg2(n)=sgsgg2(n)*adh
        sgsgg3(n)=sgsgg3(n)*adh
        sgsgg4(n)=sgsgg4(n)*adh
        sgsgg5(n)=sgsgg5(n)*adh
 275   continue
      endif
*********************************************************
************************
* Provide formatted output to the output file 'collapse.dat'
*
      write(11,400)
      write(11,410) name
      if(adh.ne.0) write(11,405)
      write(11,400)
      write(11,470)
      write(11,415)(ic(n),n=1,nbg)
      write(11,470)
      write(*,*)'SCATTER MATRIX? INSCATTER=1/ OUTSCATTER=2.'
      read(*,*) sm
      if(sm.eq.2)then
      do 280 n=1,nbg
       write(11,420) n
       write(11,470)
       write(11,430)
       write(11,440) nusigf(n),sigt(n),siga(n)
       write(11,470)
       write(11,450)
       write(11,470)
       write(11,460) sgsgg(n),sgsgg1(n),sgsgg2(n)
       write(11,470)
       write(11,460) sgsgg3(n),sgsgg4(n),sgsgg5(n)
       write(11,470)
 280  continue
```

```
      elseif(sm.eq.1)then
      do 290 n=1,nbg
       write(11,420) n
       write(11,470)
       write(11,430)
       write(11,440) nusigf(n),sigt(n),siga(n)
       write(11,470)
       write(11,455)
       write(11,470)
       if(n.eq.1)write(11,460) sgsgg(n)
       if(n.eq.2)write(11,460) sgsgg(n),sgsgg1(n-1)
       if(n.eq.3)write(11,460) sgsgg(n),sgsgg1(n-1),sgsgg2(n-2)
       if(n.eq.4)then
        write(11,460) sgsgg(n),sgsgg1(n-1),sgsgg2(n-2)
        write(11,470)
        write(11,460) sgsgg3(n-3)
       elseif(n.eq.5)then
        write(11,460) sgsgg(n),sgsgg1(n-1),sgsgg2(n-2)
        write(11,470)
        write(11,460) sgsgg3(n-3),sgsgg4(n-4)
       elseif(n.ge.6)then
        write(11,460) sgsgg(n),sgsgg1(n-1),sgsgg2(n-2)
        write(11,470)
        write(11,460) sgsgg3(n-3),sgsgg4(n-4),sgsgg5(n-5)
       endif
       write(11,470)
 290  continue
      endif
*
*
 390  format(1x,'Input TWODANT defined ICOL for NBG',I3)
 400
format(5x,'*********************************************
******')
 405  format(15x,'!!!CONVERTED TO,MACRO XSECS!!!')
 410  format(15x,a)
 415  format(10x,'ICOL=',16I3)
 420
format(5x,'********************GROUP',I3,'*****************
*******')
```

```
430 format(10x,'NUSIGF',11x,'SIGT',11x,'SIGA')
440 format(6x,e15.7,3x,e15.7,3x,e15.7)
450 format(5x,'SCATTERING MATRIX; g-g, g-g+1,...g-g+5')
455 format(5x,'SCATTERING MATRIX; g--g, g--g-1,...g--g-5')
460 format(5x,3(e15.7,3x))
470 format(' ')
*
*
    stop
    end
```

Appendix 4, Flux Volume Weighting FORTRAN 77 Program.

```
* This program reads the EDTOGX.DAT file prepared by TWODANT
* and calculates the volume average flux for each material zone
* in the problem.  As is this program only works for homogeneous
* spheres.
*
*
      DIMENSION RDAVE(200),XMESH(30),IHX(30),IDCS(25)
      DIMENSION FISRT(10000),FLUX(10000,20),FLUXN(10000,20)
      DIMENSION IHXZ(5),DELTA(5),V(200),SUMV(5)
      DIMENSION VF(200,20),SUMVF(5,20),VAFLUX(5,20)
      DIMENSION IC(16),ICOL(16),SFLUX(10000,20)
      DIMENSION HTITLE(10,10)
      CHARACTER*40 NAME
      OPEN(UNIT=4,FILE='EDTOGX.DAT',STATUS='OLD')
      OPEN(UNIT=5,FILE='VAFLUX.DAT',STATUS='NEW')
      OPEN(UNIT=6,FILE='FLUX.PLT',STATUS='NEW')
* Read in data from EDTOGX.DAT [reference LA-9184-M,rev; app. C].
*
      READ(4,*) NTITLE
*     DIMENSION HTITLE(10,NTITLE)
      DO 10 N=1,NTITLE
      READ(4,20)(HTITLE(I,N),I=1,10)
   10 CONTINUE
   20 FORMAT(20A4)
*
      READ(4,*) IDIMEN,ISADJ,NGROUP,IM,IT,JM,JT,NDUM1,NDUM2,
     +          IFISS,IGEOM
*
*     DIMENSION RDAVE(IT)
      READ(4,400) (RDAVE(N),N=1,IT)
*
      READ(4,410)(IHX(I),I=1,IM)
*
      READ(4,400) (XMESH(IZ),IZ=1,IM+1)
*
      READ(4,410)(IDCS(I),I=1,IM*JM)
*
      IF(IFISS.GT.0) READ(4,400)(FISRT(I),I=1,IT*JT)
```

```
*
*    DIMENSION FLUX(ITJT,NGROUP)
     DO 60 N=1,NGROUP
     READ(4,400)(FLUX(I,N),I=1,IT*JT)
  60 CONTINUE
 400 FORMAT(6E12.5)
 410 FORMAT(12I6)
*
* Calculate volume average flux.
*
     IHXZ(0)=0
     DO 100 IZ=1,IM
       DELTA(IZ)=(XMESH(IZ+1)-XMESH(IZ))/IHX(IZ)
       IHXZ(IZ)=IHX(IZ)+IHXZ(IZ-1)
       SUMV(IZ)=0.0
       DO 105 N=1,NGROUP
         SUMVF(IZ,N)=0.0
 105   CONTINUE
 100 CONTINUE
*
     PI=3.1415927
     R1=0.0
*
     IF(IGEOM.EQ.3)THEN
       DO 110 IZ=1,IM
         DO 120 I=IHXZ(IZ-1),IHXZ(IZ)
           R2=R1+DELTA(IZ)
           if(r2.eq.0)goto 110
           V(I)=(R2**3-R1**3)*4.0*PI/3.0
           SUMV(IZ)=SUMV(IZ)+V(I)
           DO 130 N=1,NGROUP
             VF(I,N)=V(I)*FLUX(I,N)
             SUMVF(IZ,N)=SUMVF(IZ,N)+VF(I,N)
 130         CONTINUE
           R1=R2
 120       CONTINUE
         DO 140 N=1,NGROUP
           VAFLUX(IZ,N)=SUMVF(IZ,N)/SUMV(IZ)
 140       CONTINUE
 110     CONTINUE
```

```
      ENDIF
*
* Provide output to VAFLUX.DAT.
*
      DO 200 N=1,NTITLE
      WRITE(5,210)(HTITLE(I,N),I=1,10)
  200 CONTINUE
  210 FORMAT(20A4)
*
      WRITE(5,*)' '
      WRITE(5,225)
      DO 220 N=1,NGROUP
         WRITE(5,230)N,(IZ,VAFLUX(IZ,N),IZ=1,IM)
  220 CONTINUE
*
  225 FORMAT(2X,'GROUP',1X,3('ZONE',3X,'VAFLUX',4X))
  230 FORMAT(4X,I2,3X,3(I2,1X,E12.5,2X))
*
* The next part is for generating normalixed flux plots.
*
      WRITE(*,*)'DO YOU WANT TO PLOT FLUXES; 1=YES, 2=NO'
      READ(*,*) Z
      IF(Z.EQ.2)GOTO 390
      WRITE(*,*)'INPUT PROBLEM NAME INCLOSED IN APOSTROPHIES'
      READ(*,*) NAME
*
* Normalize fluxes based on the thermal group center line flux.
*
      WRITE(*,*)'DO YOU WANT TO NORMALIZE FLUXES; 1=YES, 2=NO'
      READ(*,*) NORM
      IF(NORM.NE.1)GOTO 242
      DO 235 N=1,NGROUP
       DO 240 I=1,IT
        FLUXN(I,N)=FLUX(I,N)/FLUX(1,NGROUP)
 240   CONTINUE
 235  CONTINUE
*
* The next section sums fluxes over broad groups for use in
* plotting a comparison with collapsed group fluxes.
*
```

```
242  WRITE(*,*)'DO YOU WANT TO SUM FLUXES INTO BROAD GROUPS;
   + 1=YES, 2=NO'
    READ(*,*) SUM
    IF(SUM.EQ.2)GOTO 295
245  WRITE(*,*)'HOW MANY BROAD GROUPS?'
    READ(*,*) NBG
    WRITE(*,*)'NUMBER OF GROUPS IN EACH BROAD GROUP [ICOL]?'
    READ(*,*)(IC(N),N=1,NBG)
    ICO=0.0
    DO 250 N=1,NBG
     ICO=ICO+IC(N)
     ICOL(N)=ICO
250  CONTINUE
    IF(ICO.NE.NGROUP)THEN
     WRITE(*,*)'SUM OF ICOL.NE.NGROUP!'
     GOTO 245
    ENDIF
* Sum fluxes over the broad groups.
    DO 255 I=1,IT
     DO 260 N=1,NBG
      SFLUX(I,N)=0.0
      IF(N.EQ.1)THEN
       DO 265 J=1,ICOL(N)
        SFLUX(I,N)=SFLUX(I,N)+FLUX(I,J)
265        CONTINUE
      ELSE
       DO 270 J=ICOL(N-1)+1,ICOL(N)
        SFLUX(I,N)=SFLUX(I,N)+FLUX(I,J)
270        CONTINUE
      ENDIF
260   CONTINUE
255  CONTINUE
* Normalize summed broad group fluxes based on "thermal group".
    IF(NORM.NE.1)GOTO 282
    DO 275 N=1,NBG
     DO 280 I=1,IT
      FLUXN(I,N)=SFLUX(I,N)/SFLUX(1,NBG)
280   CONTINUE
275  CONTINUE
282  NGROUP=NBG
```

```
*
* Provide output to FLUX.PLT for use in generating a telegraph
* plot of the group fluxes.
 295 WRITE(6,*)'gen a x numbered y numbered plot.'
     WRITE(6,*)'page layout = vrh.'
     WRITE(6,*)'title is "',NAME,'".'
     WRITE(6,*)'x axis label is "RADIUS [cm]".'
     WRITE(6,*)'y axis label is "FLUX".'
     DO 300 N=1,NGROUP
      IF(N.LT.16)THEN
       WRITE(6,350) N,N,N
      ELSE
       WRITE(6,360) N,N
      ENDIF
 300 CONTINUE
     DO 310 N=1,NGROUP
      IF(NORM.EQ.1) WRITE(6,370) N,N,RDAVE(N),FLUXN(N,N)
      IF(NORM.NE.1) WRITE(6,370) N,N,RDAVE(N),FLUX(N,N)
 310 CONTINUE
     WRITE(6,*)'input data.'
     DO 320 N=1,NGROUP
      WRITE(6,380) N
      DO 330 I=1,IT
       IF(NORM.EQ.1) WRITE(6,*)RDAVE(I),',',FLUXN(I,N)
       IF(NORM.NE.1) WRITE(6,*)RDAVE(I),',',FLUX(I,N)
 330  CONTINUE
 320 CONTINUE
     WRITE(6,*)'eod.'
     WRITE(6,*)'go.'
 350 FORMAT(' curve',I2,' texture',I2,', interpolation smooth,
    + delta .0003, symbol type',I2,'.')
 360 FORMAT(' curve',I2,' texture 1, interpolation smooth,
    + delta .0003, symbol type',I2,'.')
 370 FORMAT(' message ',I2,', text "',I2,'", x=',F10.7,
    +' y=',F10.7,', in coordinate units.')
 380 FORMAT(' "group ',I2,'"')
 390 END
```

REFERENCES

1. H. C. Paxton and N. L. Pruvost, "Critical Dimensions of Systems Containing ^{235}U, ^{239}Pu, and ^{233}U," Los Alamos National Laboratory report LA-10860-MS (July 1987).

2. D. I. Garber and R. R. Kinsey, "Neutron Cross Sections," Volume II, 3rd Edition, Brookhaven National Laboratory report BNL-325 (June 1976).

3. J. J. Duderstadt and L. J. Hamilton, *Nuclear Reactor Analysis*, John Wiley & Sons, Inc., New York (1976).

4. M. J. Stanely, "Two Group Constants for Reactor Materials," APEX-369 (1976).

5. R .D. O'Dell, F. W. Brinkley,Jr., and D. R. Marr, "User's Manual for ONEDANT: A Code Package for One-Dimensional, Diffusion-Accelerated, Neutral-Particle Transport," Los Alamos National Laboratory report LA-9184-M (December 1989).

6. R. E. Alcouffe, F. W. Brinkley, D. R. Marr, and R. D. O'Dell, "User's Manual for TWODANT: A Code Package for Two-Dimensional, Diffusion-Accelerated, Neutral-Particle Transport," Los Alamos National Laboratory report LA-10049-M (February 1989).

7. L. D. Connolly, et al., "Los Alamos Group-Averaged Cross Section," Los Alamos Scientific Laboratory report LAMS-2941 (July 1963).

8. J. R. Lamarsh, *Introduction to Nuclear Engineering*, Addison-Wesley Publishing Company, Reading, Mass. (1975).

9. D. F. Cronin, "Critical Mass Studies, Part X, Uranium of Intermediate Enrichment," Oak Ridge National Laboratory report ORNL-2968 (October 1960).

10. H. K. Clark, "Subcritical Limits for Uranium-235 Systems," *Nuclear Science and Engineering* 81, 351-378 (1982).

11. S. J. Raffety and J. T. Mihalczo, "Homogeneous Critical Assemblies of 2 and 3% Uranium-235-Enriched Uranium in Paraffin," *Nuclear Science and Engineering* 48, 433-443 (1972).

12. J. C. Smith, A. V. Parker, J. G. Walford, and C. White, "Criticality of 30% Enriched Uranium Solutions in Cylindrical Geometry," Dounreay Experimental Reactor Establishment report DEG-MEMO-663 (March 1960).